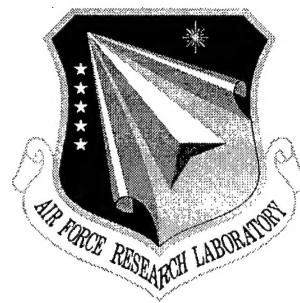


AFRL-IF-RS-TR-1999-57
Final Technical Report
April 1999



SPEAKER COUNT DETERMINATION

T-NETIX, Inc.

Manish Sharma, Yelena Kogan, Ravi Ramachandran, and Yufeng Li

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

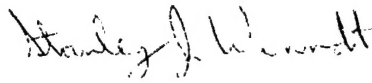
19990524 049

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

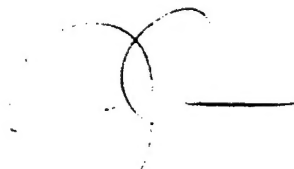
This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-1999-57 has been reviewed and is approved for publication.

APPROVED:



STANLEY J. WENNDT
Project Engineer



FOR THE DIRECTOR:

JOSEPH CAMERA, Deputy Chief
Information & Intelligence Exploitation Division
Information Directorate

If your address has changed or if you wish to be removed from the Air Force Research Laboratory Rome Research Site mailing list, or if the addressee is no longer employed by your organization, please notify AFRL/IFEC, 32 Brooks Road, Rome, NY 13441-4114. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
<small>Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.</small>				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE April 1999	3. REPORT TYPE AND DATES COVERED Final Mar 97 - Nov 98		
4. TITLE AND SUBTITLE SPEAKER COUNT DETERMINATION		5. FUNDING NUMBERS C - F30602-97-C-0046 PE - 35885G PR - 3188 TA - CO WU - 03		
6. AUTHOR(S) Manish Sharma, Yelena Kogan, Ravi Ramachandran, and Yufeng Li				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) T-NETIX, Inc. 371 Hoes Lane, Suite 203 Piscataway NJ		8. PERFORMING ORGANIZATION REPORT NUMBER N/A		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratory/IFEC 32 Brooks Road Rome NY 13441-4114		10. SPONSORING/MONITORING AGENCY REPORT NUMBER AFRL-IF-RS-TR-1999-57		
11. SUPPLEMENTARY NOTES Air Force Research Laboratory Project Engineer: Stanley J. Wenndt/IFEC/(315) 330-7244				
12a. DISTRIBUTION AVAILABILITY STATEMENT Approved for public release; distribution unlimited.		12b. DISTRIBUTION CODE		
13. ABSTRACT (Maximum 200 words) Speaker count is the process of automatically identifying segments of speech that contain multiple speakers before attempting to apply any co-channel interference reduction schemes. By computing the variance of the pitch estimate, a heuristic approach is used to classify a 30 msec frame of speech as being from a single talker or from multiple talkers. Using TIMIT data, 79% of the frames were classified correctly.				
14. SUBJECT TERMS Speaker Count, Multiple Speaker Determination		15. NUMBER OF PAGES 96		
		16. PRICE CODE		
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

Table of Contents

1. INTRODUCTION.....	4
1.1. OBJECTIVE	4
1.2. SCOPE OF THE REPORT.....	4
1.3. PROBLEM DEFINITION	4
1.4. SYSTEM OVERVIEW	6
2. PATTERN CLASSIFIERS	10
2.1. VECTOR QUANTIZER CLASSIFIER.....	11
2.2. NEURAL TREE NETWORK	12
2.3. GAUSSIAN MIXTURE MODEL (GMM)	13
3. LINEAR PREDICTION-BASED SPECTRAL FEATURES.....	14
3.1. LINEAR PREDICTION (LP) AND CEPSTRUM BASED FEATURES	14
3.1.1. <i>Linear Prediction Coding Coefficient (LPCC)</i>	15
3.1.2. <i>Area Coefficients</i>	16
3.1.3. <i>Reflection Coefficients</i>	16
3.1.4. <i>Log Area Ratios</i>	16
3.1.5. <i>Line Spectral Frequencies</i>	16
3.1.6. <i>Linear Prediction Cepstrum</i>	17
3.1.7. <i>Bandpass Filtered and Quefrency Weighted Cepstrum</i>	18
3.1.8. <i>Mel Frequency cepstrum coefficient Analysis</i>	19
3.1.9. <i>Adaptive Component Weighted Cepstrum</i>	20
3.1.10. <i>Postfilter Cepstrum</i>	22
3.1.11. <i>Feature Fusion</i>	22
3.2. EXPERIMENTAL SETUP	23
3.2.1. <i>Speech Database</i>	23
3.2.2. <i>Baseline Experimental Setup</i>	23
3.2.3. <i>Performance Metric and Decision Error Classification</i>	24
3.3. EXPERIMENTAL RESULTS	24
3.3.1. <i>Comparing the Features Performance with Different Classifiers</i>	25
3.3.1.1. <i>System Performance with Vector Quantizer (VQ) Classifier</i>	25
3.3.1.2. <i>System Performance with Neural Tree Network (NTN) Classifier</i>	26
3.3.1.3. <i>System Performance with Gaussian Mixture Model (GMM) Classifier</i>	26
3.3.2. <i>Adding Voicing Detection to Spectral Feature Algorithms</i>	27
3.3.3. <i>Post-processing of Final Decision Labels Using Smoothing</i>	29
3.3.4. <i>Utterance Level versus Frame Based Performance</i>	31
3.3.5. <i>Transitional Frames Error Analysis</i>	31
3.3.6. <i>System Performance with Non-silence Speech</i>	33
3.3.7. <i>System Performance at Various Signal-to-Interference Ratio (SIR)</i>	34
3.3.8. <i>System Performance with Increased Number of Speaker Population</i>	35
3.3.9. <i>Correlation between System Decision and Frame Energy</i>	36
3.3.10. <i>System Performance on an Increased Number of Simultaneously Talking Speakers</i>	38
3.3.11. <i>Feature Fusion</i>	40
3.3.11.1. <i>Fusion of the Features with Complementary Error Distributions</i>	40
3.3.11.2. <i>Fusion of the LP Cepstrum and Short-term Variance (delta) Features</i>	42
3.3.12. <i>Comparing Performance of the Speaker Count System on Different Speaker Gender Combinations</i>	43

3.3.13. Dependence of System Performance on a Processing Frame Size.....	44
3.3.14. New Approach for Speech-silence Discrimination Error.....	45
3.3.15. New Silence Detection Algorithm.....	46
4. PITCH BASED METHODS	48
4.1. PITCH PREDICTION FEATURE (PPF).....	48
4.2. LPC RESIDUAL BASED PITCH FEATURE.....	53
4.2.1. Typical case	56
4.2.2. Pitch Doubling.....	58
4.2.3. Pitch Halving	59
4.2.4. Female speaker dominating.....	60
4.2.5. Male speaker dominating.....	62
4.2.6. Analysis of Energy and Energy Ratio	64
4.3. EXPERIMENTAL SETUP	66
4.4. EXPERIMENTAL RESULTS FOR PPF.....	66
4.4.1. System Performance with Vector Quantizer Classifier.....	66
4.4.2. System Performance with Neural Tree Network.....	67
4.4.3. System Performance on an Increased Number of Simultaneously Talking Speakers...	67
4.5. EXPERIMENTAL RESULTS FOR LPC RESIDUAL BASED PITCH.....	68
4.5.1. System Performance with Baseline Parameters	68
4.5.2. Adding Voicing Detection to LPC Residual Based Pitch Algorithm	69
4.5.3. System Performance with Non-silence Speech	69
4.5.4. System Performance at Various Signal-to-Interference Ratios (SIR).....	70
4.5.5. System Performance on an Increased Number of Simultaneously Talking Speakers...	72
4.5.6. Study Correlation between System Decision and Frame Energy	72
4.5.7. Comparing Performance of the Speaker Count System on Different Speaker Gender Combinations	75
4.5.8. New Approach for Speech-Silence Discrimination Error.....	76
4.5.9. New Silence Detection Algorithm.....	77
4.5.10. System Performance Different Pitch Variance Threshold.....	77
4.5.11. Classification of Low Energy Frames as Single Speaker Frames	79
5. EXPERIMENTS ON GREEN FLAG DATABASE	81
5.1. System Performance Benchmarks.....	81
5.2. System Performance Using Different Pitch Variance Threshold	82
6. SUMMARY AND CONCLUSION.....	84
6.1. KEY OBSERVATIONS.....	84
6.2. FUTURE RESEARCH.....	84
REFERENCES.....	85

1. Introduction

1.1. Objective

The fundamental objective of the *Speaker Count Determination for Co-channel Interference Reduction* investigation is to analyze a composite or co-channel speech signal consisting of the voices of multiple speakers, and determine on a short time speech segment-by-segment (10-100 milliseconds) basis, the number of speakers talking.

1.2. Scope of the Report

This chapter (Chapter 1) introduces the problem and the objective of this investigation, describes the organization of the report, provides background information on methods and techniques used in the project, and summarizes the overall architecture of a *co-channel speaker count determination* system.

Chapter 2 discusses the choices of pattern classifiers used to perform the task: Vector-Quantizer (VQ) classifier, Neural Tree Network (NTN), Gaussian Mixture Model (GMM).

The above pattern classifiers were evaluated along with different discriminative speech features (characteristics) in experiments described in chapters 3 and 4.

Chapter 3 describes a speaker count determination system using Linear Prediction (LP) based spectral speech features.

It defines the baseline experimental setup conditions and system decision errors. The chapter provides the results obtained in various experiments investigating the performance of proposed Linear Prediction (LP) based speech spectral features using different classifiers.

Chapter 4 describes an alternate embodiment of the speaker count determination system using pitch based speech analysis techniques. Pitch prediction feature (PPF) and Linear Prediction Coding (LPC) residual based pitch variance algorithms were tested. The experimental results are provided for each of these features.

Chapter 5 provides experimental results and analysis on the "Green Flag" speech corpus supplied by the contracting agency (AFRL-Rome). Most of the proposed system embodiments in Chapter 3 and 4 were evaluated.

Chapter 6 is a summary of the current investigation. It brings up the successful approaches as directions for future work.

1.3. Problem Definition

Speaker count determination is an important first step in achieving co-channel interference reduction and co-channel speaker separation. Co-channel interference adversely affects the performance of computer-automated speech processing systems, such as speaker identification and speech recognition systems. Obviously, it would be desirable to suppress the interference due to the

interfering speakers, so that the speech of the target speaker can be more easily understood, either by a human listener or a by machine. Co-channel interference reduction reduces, or even removes, the interfering voices of speakers other than the speaker of concern (target speaker).

Co-channel interference may be introduced by different sources and can greatly decrease the intelligibility of the target speaker's voice. Co-channel interference reduction is critically important for certain communication applications. In other applications it is desirable to recover the speech of not only the target speaker, but that of the interfering speakers as well, i.e. to separate speech of several speakers. Co-channel *speaker separation* refers to the recovery of the individual constituent speech signals present in a co-channel signal, which consists of the superposition of the voices of multiple persons speaking simultaneously on the same monophonic communications channel.

Over the past years the general approach to solving the problem of co-channel speaker separation and co-channel interference reduction have been to apply signal processing techniques directly to the co-channel speech without any pre-processing. Such methods as harmonic magnitude selection [Parsons76], harmonic magnitude suppression [Hanson84], spectral sampling [Min88], adaptive comb filtering [Frasier76], pitch and spectral envelope tracking [Zissman92] and others were applied. The failure of these approaches to deliver accurate results cannot be understated.

Speaker count determination is the problem of automatically identifying the number of speakers present in co-channel speech before separation. This approach was never directly defined in engineering literature, but a similar problem called Automatic Talker Activity Labeling has been reported in [Zissman90][Zissman91]. In these works, co-channel speech was used as input where frames were labeled either target (primary speaker), jammer (interfering speaker) or talker-jammer (co-channel speech). A classifier was then used to train front-end features vector for the "target" speaker, the "jammer" speaker and the combination of both speakers. During the recognition, the detector was presented with speech from the target, jammer and combination of target-jammer. The detector's task was to use the stored references to identify which of the three possible sources produced the input and report that result. The detectors were then evaluated on its ability to label the test input correctly. With this approach, and the use of the mel-cepstrum feature, 80% correct detection rate was recorded. The disadvantage of the Automatic Talker Activity Labeling algorithm is that it looks at the narrow problem of a closed set of speakers, i.e. the same set of speakers were used to generate training and testing co-channel speech data.

In the current work, alternate forms of Speaker Count Determination algorithms are investigated. In developing a practical speaker count algorithm there are some considerations and limitations that must be taken into account. The ideal algorithm should be able to estimate the number of speakers in a co-channel speech segment. There are several considerations that need to be accounted for. The co-channel signal is quite similar to individual speech signals, making it very difficult to identify a co-channel speech. There also exists a

question of the theoretical limit on the performance of such an algorithm, based on whether it is possible to transmit an arbitrary number of independent signals, each having bandwidth W , along a communication channel of bandwidth W , thereby violating some of the basic laws of communications and information theory. Despite these limitations, there are considerations that allow for a degree of success with these algorithms.

The speaker count determination algorithm can be evaluated with closed or open set of speakers. That means, similar speech data from the target and jammer speakers may be available (closed set) or not available (open set) *a priori* to testing the algorithm. The algorithm proposed here can be described as an “open-set” evaluation, since no speech data from the test speakers was available during training. Moreover, the utterances of the primary and interfering speakers are text-independent. This makes the problem more difficult than the one described in [Zissman90]. However, it makes it a more practical one. In reality, most speaker processing applications are “open-set” and text-independent in nature.

The solution of speaker count determination problem becomes the first step in the building a robust and accurate algorithm for a co-channel interference reduction and the separation of co-channel speech. Once the overall speech processing system detects that the signal has been corrupted by interfering voices, the speaker may be requested to retransmit, or a co-channel speaker separation system may “kick in”. The end-result is that such system will become more practical and robust, since it can automatically identify when the co-channel interference has occurred.

Speaker count determination task is highly significant for speaker and language identification systems, and will be extremely useful for tactical voice communication systems. In those systems interference due to unwanted speakers is of great concern, since such interference often impairs both the intelligibility and identification of the target speaker's voice.

1.4. System Overview

An overview of a speaker count determination system is described below. The task is to establish a speaker count on a short time frame-by-frame basis in which the frame size is very small (30 milliseconds). Such a frame size is needed to achieve effective speaker separation based on pitch and spectral envelope methods.

There are three different types of speech segments found in a co-channel speech, namely, no speakers talking (silence), one speaker talking, or multiple (more than one) speakers talking. Based on energy thresholding, segments (frames) are labeled as either silence or speech. Parametric representation of speech frames is used as discriminative features. Discriminative features for the frames consisting of speech (non-silence) only are utilized. Developing “good” discriminative features, that clearly indicate the number of speakers present, is one of the key tasks. In a pattern classification based system approach, training procedure is required. During the training phase the feature vectors are used to

train a pattern classifier. The trained pattern classifier is then used during testing phase to establish a decision regarding the speaker count on frame by frame basis. Alternately, in a rule-based system approach an explicit training stage is not required. Instead a decision rule is formulated by evaluating and studying some sample cases.

The system, therefore, is comprised of several modules, and the successful implementation of each of them contributes to the final system performance. The most important building blocks of the system are:

- speech-silence detection mechanism;
- discriminative features to characterize speech information;
- pattern classifier to make decision about the number of speakers present.

The experiments were conducted on speech utterances from the standard TIMIT¹ speech corpus, which consists of single-speaker recordings. Multi-speaker or co-channel speech was simulated by mixing single-speaker utterances in a sample-by-sample additive manner. Each of the single-speaker recordings was passed through a silence detector, which partitioned the recording into contiguous, equal-sized frames and marked each frame as either silence or non-silence. The speaker count for each frame of the mixed speech utterance was determined as follows:

```
IF ((frame 1 is silence) AND (frame 2 is silence))
    speaker count = 0
ELSE IF ((frame 1 is silence) AND (frame 2 is non-silence))
    speaker count = 1
ELSE IF ((frame 1 is non-silence) AND (frame 2 is silence))
    speaker count = 1
ELSE IF ((frame 1 is non-silence) AND (frame 2 is non-silence))
    speaker count = 2
```

Various techniques researched during the project will be described in the remainder of the report. The algorithm of the most successful combination of methods is as follows:

Label speech frames (silence detection);

Obtain the linear predictive (LP) cepstrum [Rabiner78][Rabiner93] feature vectors; select feature vectors for all non-silence, i.e. speech, frames;

Training: Cluster feature vectors from one speaker and multi-speaker speech frames separately to design vector quantizer (VQ) codebooks using the Linde-Buzo-Gray (LBG) [Linde80] algorithm;

Testing: Find the closest VQ codebook to a test feature vector in order to establish a decision on a frame by frame basis regarding the speaker count.

During testing, the "correct" reference labels for each processing frame is created. They serve as the ground truth to compare against the decision estimated by the system. The system performance is evaluated as the

¹ TIMIT speech corpus can be obtained from Linguistic Data Consortium, University of Pennsylvania, Philadelphia, USA, (215)-898-0464, Fax: (215)-573-2175, URL: <http://www ldc.upenn.edu>

percentage of total number of test frames whose speaker count was identified correctly.

The general scheme for training and testing procedures for Speaker Count system is shown in Figures 1.1 and 1.2.

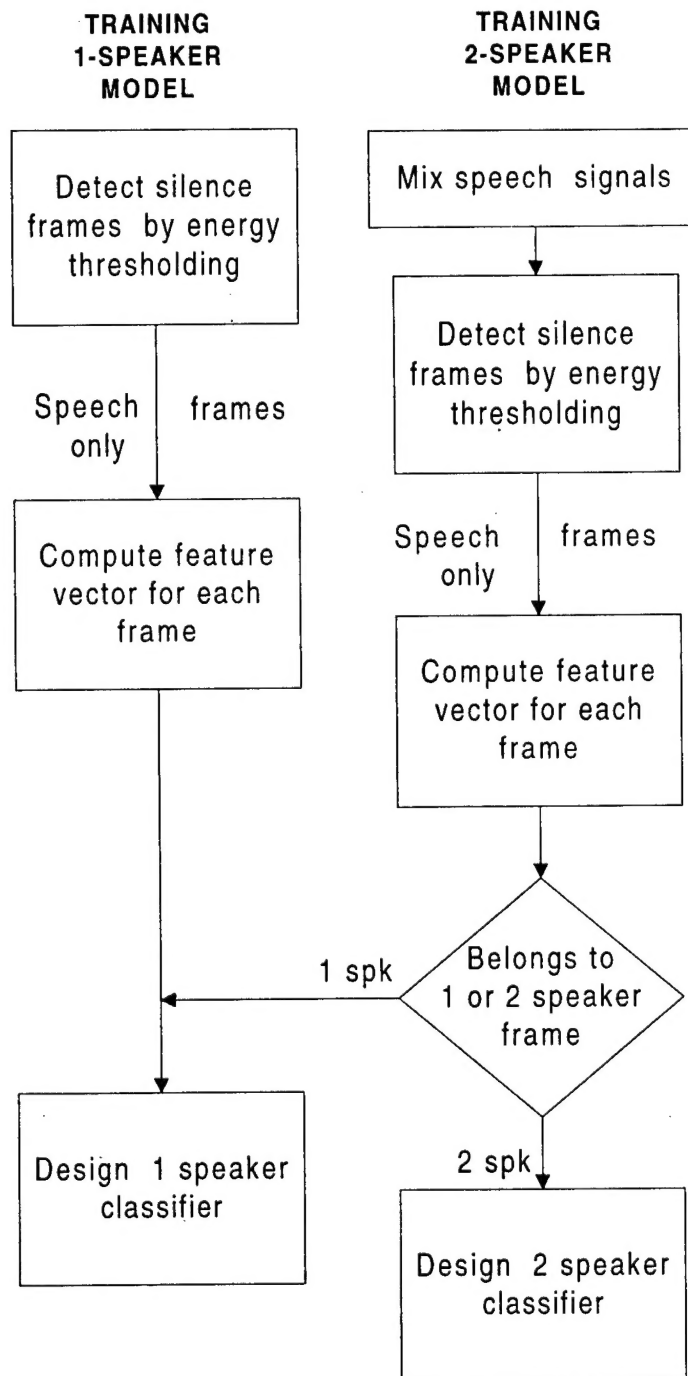


Figure 1.1 The general scheme for training procedures for Speaker Count System

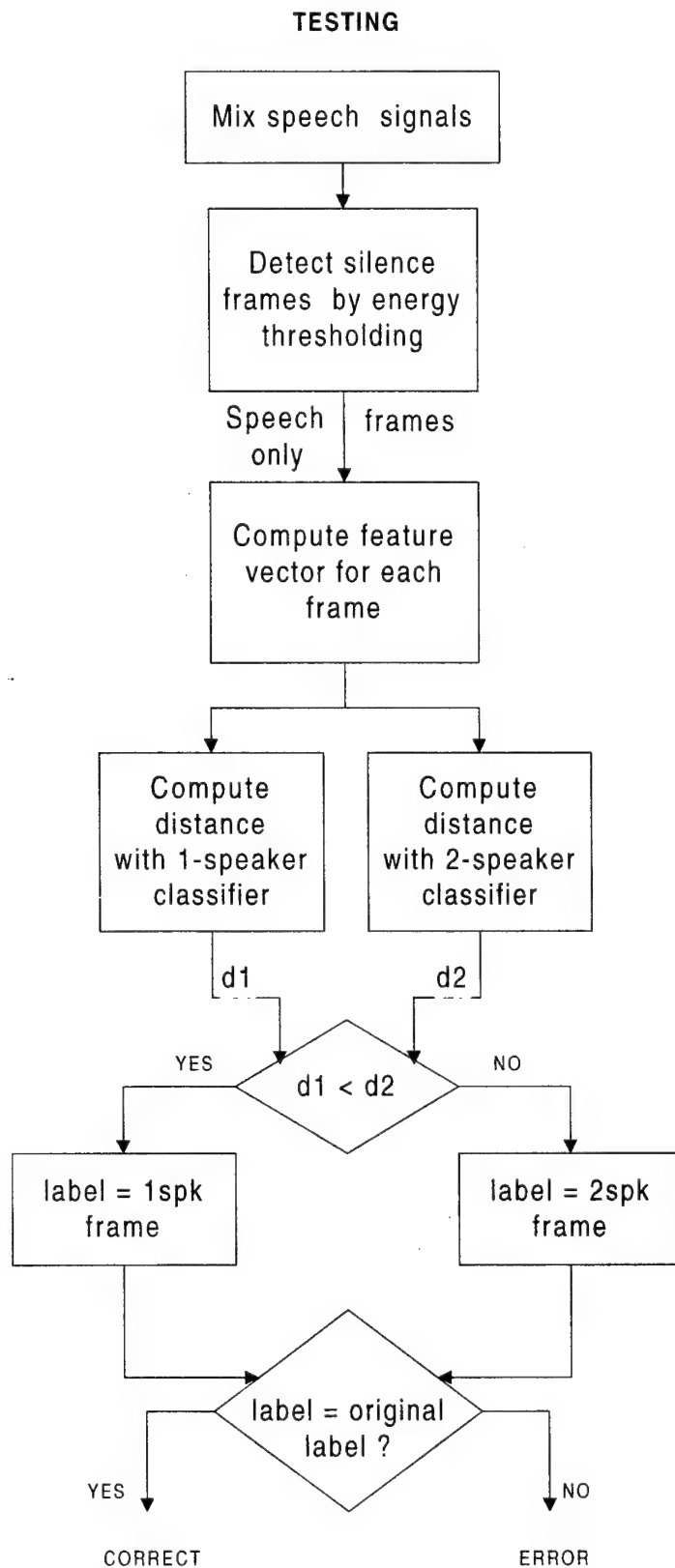


Figure 1.2 The general scheme for testing procedures for Speaker Count System

2. Pattern Classifiers

Pattern classification techniques involve assignment of an observed event to one of several pre-specified categories. An extensive study of classification problems have led to several mathematical models that provide a theoretical basis for pattern classifier design. A pattern classifier evaluates the evidence presented to it and makes a decision about the category type of the unknown input.

In *speaker recognition systems*, vector quantizers (VQ) and neural tree network (NTN) classifiers have been used successfully to render decisions about the identity of a speaker [Farrell94][Rosenberg87] among a group of N speakers. Each speaker is represented by a VQ codebook or NTN model, which is configured during training. During testing, the feature vectors are obtained from one utterance consisting of many frames. These feature vectors are applied to each of the VQ codebook or NTN models (depending on which classifier is used) to get N distinct scores. The model with the best score identifies the speaker.

The *speaker count determination* problem is slightly different in that models are required to represent single speaker and multi-speaker speech characteristics. Also, in contrast to speaker recognition, a decision has to be made for each individual frame rather than for an entire utterance. The speaker count is determined for each frame and hence, the decision is taken using only one feature vector. The general pattern classification scheme for speaker count is shown in Figure 2.1. Two VQ codebooks are developed from the training feature vectors, each dedicated to one of the two types of speech conditions (single speaker or multi-speaker) encountered. This is known as unsupervised learning in that training data pertaining to another condition does not influence the codebook for a particular condition.

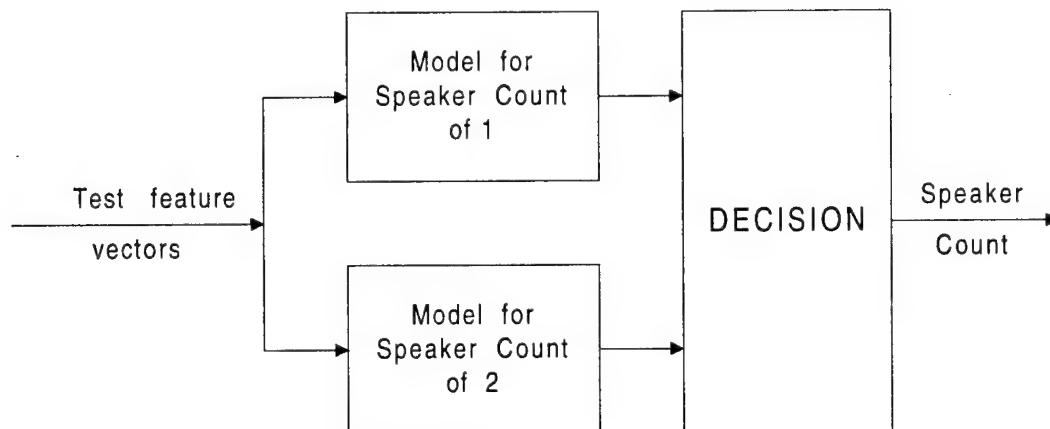


Figure 2. The general scheme for speaker count

The NTN classifier [Sankar93] is a hierarchical classifier that combines the properties of decision trees and feed-forward neural networks [Farrell94]. The NTN uses the tree architecture to implement a sequential linear decision strategy [Duda73]. The architecture of the NTN is determined during training. Thus, it is self-organizing. Also, NTN training is supervised in that the training data from other categories may also be presented when training the model for one category.

Gaussian Mixture Model (GMM) is a parametric classifier that uses a set Gaussian (Normal) probability density functions to approximate the probability density function of given pattern vectors. GMM works similar to VQ, but provides an extra information of co-variance for each cluster of analyzed data. The GMM can be mathematically represented as:

$$f(x) = \sum_{i=1}^{i=M} w_i N(x, \mu_i, \Sigma_i),$$

where $f(.)$ is the GMM probability density function for vector x ,

M is the total number of Gaussian mixtures in the model,

w_i is the weight or mixture coefficient for the i -th Gaussian component,

$N(.)$ is the i -th Gaussian probability component with mean vector μ_i and co-variance matrix Σ_i

The mixture weights w_i sum up to one. For a sufficiently large number of mixture densities, M , the GMM can be used to arbitrarily accurately approximate any continuous probability density function.

2.1. Vector Quantizer Classifier

Feature vector is computed for each frame of the co-channel recording. The feature vectors corresponding to those frames containing one speaker are grouped together and the feature vectors corresponding to all those frames containing two speakers are grouped together. Each of these groups of feature vectors is used to train a vector quantizer classifier and generate a corresponding vector quantizer codebook. A codebook for each category is designed by the Linde-Buzo-Gray (LBG) algorithm [Linde80]. Each of the codebooks has the same size or number of code vectors. The feature vectors corresponding to frames with zero speakers (silence) are not used to generate a vector quantizer codebook.

Given a test feature vector obtained from a particular speech frame, a VQ classifier is used to establish the speaker count. Each test vector is classified as single-speaker or multi-speaker by computing the squared Euclidean or L_2 distance of the test feature vector to each of the codewords in the single-speaker codebook and each of the codewords in the multi-speaker codebook. Hence, two distances are recorded, minimum codeword distance for each codebook. The

feature vector is classified as belonging to that class (single-speaker or multi-speaker) for which the distance to the closest codeword in the corresponding codebook is minimum. The codebook that renders the smallest distance identifies the speech condition. This method is very similar to the VQ based classifier used in speaker recognition [Rosenberg87].

2.2. Neural Tree Network

In the NTN classifier, each training feature vector has a label indicating the category it represents. Each node at every level of the NTN divides the input training vectors into a number of exclusive subsets of the training data. If the entire set of training data at a particular node is of the same class or condition (has the same label), then that node becomes a leaf. Otherwise, the data is split into several subsets, and are given to the children of this node for further classification. This procedure is repeated until all the data is completely uniform at the leaf nodes. The leaf nodes of the NTN partition the feature space into homogeneous subsets, meaning a single class at each leaf node. An illustration of the concept is given in Figure 2.2. The training data comes from two classes labeled as 0 and 1. The circles represent nodes and the squares represent leaves. The nodes can be thought of as being hyperplanes that partition the space into exclusive subspaces. These subspaces are further partitioned until a leaf is reached.

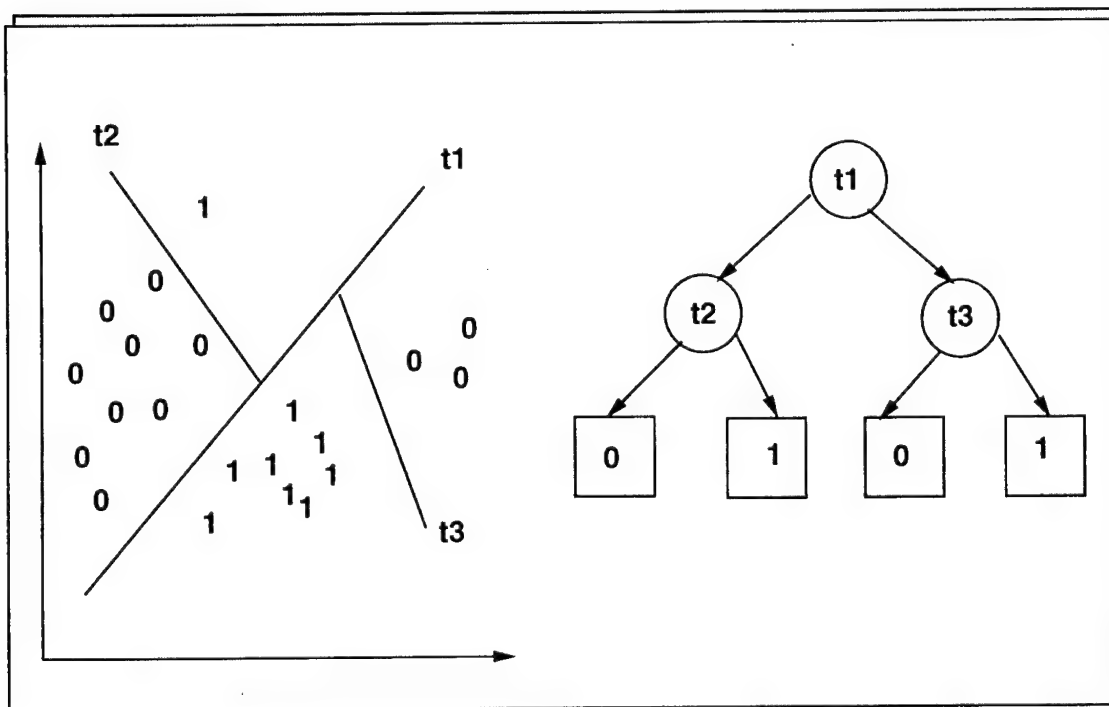


Figure 2.2. Concept of Neural Tree Network. The circles represent internal nodes and the squares represent terminal leaves.

For the Speaker Count Determination system the NTN is grown from training data consisting of two labels, namely, single-speaker and multi-speaker speech. During testing, the test feature vector is evaluated through the NTN until it reaches a particular leaf. The label assigned to the leaf classifies the speech frame.

2.3. Gaussian Mixture Model (GMM)

During training phase, the feature vectors corresponding to those frames containing one speaker are grouped together and the feature vectors corresponding to all those frames containing two speakers are grouped together. Each of these groups of feature vectors is used to train a GMM, i.e., to estimate the mean, co-variance and mixture weight of each of the mixture components in the GMM.

Given a test feature vector obtained from a particular speech frame, the pre-trained GMM classifiers are used to establish the speaker count. The feature vector is classified as belonging to that class (single-speaker or multi-speaker) for which the corresponding GMM probability score is maximum.

3. Linear Prediction-based Spectral Features

The procedure of transforming the original speech waveform signal into a domain of meaningful features is the first and very important task in any speech processing system. Correctly chosen feature will capture the necessary characteristics of speech and suppress those of non-importance.

Various types of spectral features were investigated to choose the feature that would optimize the system performance. The next section gives the review of the Linear Predictive (LP) and cepstral based speech feature analysis. This discussion is followed by a summary of all the experimental results obtained by using these features.

3.1. Linear Prediction (LP) and Cepstrum based Features

Linear prediction (LP) analysis is a very popular technique in speech analysis. Its popularity stems from its compact, yet precise representation of the speech spectra, as well as its relative simplicity of computation. The motivation of using the LP based features is that they have been shown to be highly useful for speech and speaker recognition applications [Atal74]. Cepstral coefficients based methods have been successfully used in speech recognition applications [Oppenheimer89] because of their ability to represent the spectral envelope while being not sensitive to the pitch and phase.

In the current study the following spectral features based on Linear prediction and cepstral analysis were examined:

1. Linear Prediction Coding Coefficient (LPCC);
2. Area Coefficients (Area);
3. Reflection Coefficients (Refl);
4. Log Area Ratios (LAR);
5. Line Spectral Frequencies (LSF);
6. Linear Prediction Cepstrum (LPcep);
7. Bandpass Liftered Cepstrum (BPLcep);
8. Quefrency Weighted Cepstrum (Qcep) ;
9. Adaptive Component Weighting Cepstrum (ACWCep);
10. Postfilter Cepstrum (PFCep);
11. Mel Frequency Cepstrum Coefficient (MFCC).

Some background theory of the above listed feature derivations is provided below. For a detailed information about these spectral features please refer to any standard text on digital speech processing [Rabiner78] [Deller93] [Rabiner93] [Markel76].

3.1.1. Linear Prediction Coding Coefficient (LPCC)

A reasonably general linear discrete-time model for speech production includes a vocal tract linear filter model $H(z)$ and a discrete time glottal excitation signal $u(n)$. During unvoiced speech activity, the excitation source is a flat spectrum noise source modeled by a random noise generator. During periods of voiced speech activity, the excitation uses an estimate of the local pitch period to set an impulse train generator that drives a glottal pulse shaping filter.

A linear predictive (LP) analysis of speech assumes an all-pole model for the vocal tract filter $H(z)$

$$H(z) = \frac{1}{1 - \sum_{i=1}^p a(i)z^{-i}} = \frac{1}{1 - A(z)}$$

The time-domain waveform signal data is specified by the difference equation:

$$s(n) = \sum_{i=1}^p a(i)s(n-i) + e(n)$$

where, $e(n)$ is the prediction error signal.

The objective of the LP algorithm is to estimate the Linear Prediction coding coefficients (LPCC) $\{a(i)\}$ associated with the all-pole system function $H(z)$. This can be achieved by solving the above difference equation to minimize the average squared prediction error [Markel76][Deller93]. Two widely used techniques to estimate LPCC are the auto-correlation method and covariance method. The auto-correlation technique was preferred in current experiments, since it produces a stable LP model (all poles within the unit circle). The Levinson-Durbin recursion algorithm was used to implement the auto-correlation technique [Markel76][Deller93]. The Levinson-Durbin recursion is a recursive-in-model-order solution for the auto-correlation equations. The solution for the desired order- p model is successively built up from lower-order models.

The magnitude spectrum of $H(z)$ approximates the speech spectrum which in turn indicates the frequency response of the vocal tract. An order of $p=12$ is usually used for telephone bandwidth speech data to get an adequate description of the vocal tract. An LP model is extremely good for the case of describing the vocal tract of one speaker. But for co-channel signals the LP model has not been experimented before. It is assumed that the LP based features for speech segments having speech from multiple speakers will show a substantial deviation from those having speech from a single speaker.

3.1.2. Area Coefficients

The multi-tube lossless model of the vocal tract represents the vocal tract as a series of concatenated lossless acoustic tubes, as shown in figure 3.1. The complete vocal tract model consists of a sequence of tubes with cross-sectional areas A_k and lengths l_k .

When sound wave propagates through the concatenated tube sections, a portion of the traveling wave in tube k is transmitted to tube $k+1$, and a portion is reflected back into tube k . The portion of the traveling wave transmitted or reflected back is dependent on the cross-sectional areas of the two tube sections.

The area coefficients can be obtained as a by product when solving for LP parameters.

3.1.3. Reflection Coefficients

The reflection coefficients can also be obtained while solving for LP parameters using auto-correlation technique and Levinson-Durbin recursion. These coefficients have close relationship to the traveling wave reflection coefficients in multi-tube model of the vocal tract.

It should be noted that the LP coefficients $\{a(i)\}$ can be obtained from reflection coefficients, and vice-versa.

3.1.4. Log Area Ratios

Log area ratio (LAR) parameters are sometimes used in place of the reflection coefficients because when a reflection coefficient has a magnitude near unity, the results are sensitive to quantization errors. The log transformation within LAR computation warps the amplitude scale of the parameters to decrease the sensitivity.

3.1.5. Line Spectral Frequencies

The Line spectral frequencies were introduced as another alternative to LP parameters for speech coding with better quantization and interpolation properties [Sugamura81].

Consider two polynomials $P(z)$ and $Q(z)$ of order $p+1$, generated from the LP polynomial $A(z)$ as follows:

$$P(z) = A(z) + z^{-(p+1)} A(z^{-1})$$

$$Q(z) = A(z) - z^{-(p+1)} A(z^{-1})$$

$P(z)$ and $Q(z)$ correspond to lossless models of the vocal tract with the glottis closed and open respectively. It can be shown that all zeros of $P(z)$ and $Q(z)$ lie on the unit circle. In fact, $P(z)$ has a real zero at $z=-1$, $Q(z)$ has a real zero at $z=1$, and all other zeros are complex and interleaved. These zeros comprise the Line Spectral Pair (LSP) parameters. Since, the zeros occur in complex conjugate pairs for both $P(z)$ and $Q(z)$, there are only p unique zeros needed to specify the model. Moreover, since the magnitude of the complex zeros is known to be unity, there is only a single parameter, i.e., the frequency or angle, needed to represent the model. The frequency representation of the LSP parameters is called the Line Spectral Frequencies (LSF).

3.1.6. Linear Prediction Cepstrum

Consider a (not necessarily causal) signal $x(n)$ whose z -transform $X(z)$ exists and has a region of convergence that includes the unit circle. Suppose $C(z) = \log(X(z))$ has a convergent power series expansion in which again, the region of convergence includes the unit circle. The cepstrum is defined as the inverse z -transform of $C(z)$ in that [Oppenheimer89]

$$C(z) = \sum_n c(n) z^{-n}.$$

Note that $c(n)$ is also not necessarily causal. Let us continue by assuming that $X(z)$ is a rational function of z that is completely described by its poles, zeros and gain. Then, the cepstrum $C(z)$ will have the following properties [Oppenheimer89]:

- The sample $c(0)$ is the natural logarithm of the gain.
- The poles and zeros of $X(z)$ inside the unit circle contribute only to the casual part of $c(n)$ starting at $n=1$.
- The poles and zeros of $X(z)$ outside the unit circle only to the anticausal part of $c(n)$.
- The cepstrum is causal if and only if $X(z)$ is minimum phase.
- The cepstrum is anticausal if and only if $X(z)$ is maximum phase.
- The cepstrum $c(n)$ decays as fast as $1/|n|$ as n approaches ∞ and $-\infty$.
- The cepstrum has infinite duration whether $x(n)$ is of finite or infinite duration.
- If $x(n)$ is real, $c(n)$ is real.

As a special case of the more general $X(z)$, consider the minimum phase all-pole linear predictive (LP) filter:

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 - \sum_{k=1}^p a(k) z^{-k}}$$

obtained by the auto-correlation method. Given that all the poles $z=z_i$ are inside the unit circle and the gain is 1, the causal LP cepstrum $c_{lp}(n)$ of $H(z)$ is given by [Rabiner93],[Oppenheimer89]:

$$c_{lp}(n) = \frac{1}{n} \sum z_i^n, \quad n > 0$$

$$c_{lp}(n) = 0, \quad n < 0$$

A recursive relation between the LP cepstrum and the predictor coefficients is given as [Rabiner93]:

$$c_{lp}(n) = a_n + \sum_{i=1}^{n-1} \frac{i}{n} c_{lp}(i) a_{n-i}$$

The use of this recursion allows for an efficient computation of $c_{lp}(n)$ and avoids polynomial factorization. Since $c_{lp}(n)$ is of infinite duration, the feature vector of dimension p consists of the components $c_{lp}(1)$ to (p) which are the most significant due to the decay of the sequence with increasing n . Even with this truncation, the mean-square difference between two LP cepstral vectors is approximately equal to the mean-square difference between the log spectra of the corresponding all-pole LP filters [Rabiner93]. Hence, this provides a good measure of the difference in the spectral envelope of the speech frames that the cepstral vectors were derived from.

The cepstral coefficients have been proven to be much more robust and reliable than other LPC features, especially in applications related to the speaker and speech recognition. In addition, there are several other modified cepstral derived coefficients.

3.1.7. Bandpass Liftered and Quefrency Weighted Cepstrum

The basic idea behind cepstral weighting is to account for the sensitivity of the low-order cepstral coefficients to overall spectral slope and the sensitivity of the high-order cepstral coefficients to noise [Rabiner93]. Weighting is accomplished by multiplying $c_{lp}(n)$ by a window $w(n)$ and using the weighted cepstrum as the feature vector. This weighting operation is also known as *liftering*. The first consequence of liftering is in extracting a finite dimensional feature vector from an infinite duration $c_{lp}(n)$. Also, careful choices of $w(n)$ enhance robustness.

There are several schemes of weighting which differ in the type of cepstral window $w(n)$ that is used. The simplest one is the rectangular window as given by:

$$w(n) = \begin{cases} 1, & n = 1, 2, \dots, L \\ 0, & \text{otherwise} \end{cases}$$

where L is the size of the window. The first L samples, which are the most significant due to the decaying property, are kept. Other forms of $w(n)$ include *quefrency liftering* (or linear weighting) where:

$$\omega(n) = \begin{cases} n, & n = 1, 2, \dots, L \\ 0, & \text{otherwise} \end{cases}$$

and *bandpass liftering* (BPL) [Rabiner93][Juang] where:

$$\omega(n) = \begin{cases} 1 + (L/2) \sin(n\pi / L), & n = 1, 2, \dots, L \\ 0, & \text{otherwise} \end{cases}$$

The quefrency liftering weights each individual cepstral component by its index n thereby downplaying the lower order components. The BPL weights a cepstral sequence by a raised sinusoidal function so that the lower and higher order components are de-emphasized. Note that the weighting schemes described are fixed in the sense that the weights are only a function of the cepstral index and have no explicit bearing on the instantaneous variations in the cepstrum that are introduced by different environmental conditions (like noise and channel effects).

3.1.8. Mel Frequency cepstrum coefficient Analysis

The perception of either pure tone sounds or speech signals by humans have been shown to follow a nonlinear scale. This has led to the definition of what is known as subjective pure tones. Thus, for every pure tone defined by actual frequency f , measured in Hz, a subjective pitch is measured on a scale called the mel or bark scale. As a standard reference, a pitch of a 1kHz tone, 40db above the hearing threshold, is defined as 1000 mels. Mathematically it has been shown that the subjective pitch in mels increases less and less rapidly as the stimulus frequency is increased linearly [Rabiner93][Stevens57]. Therefore, this means that the human auditory system has poorer discrimination at high frequencies than at low frequencies [Wang92].

Another factor that is significant and should be incorporated into a subjective model is the concept of critical bands which are basically the perceptual weighting of spectral energy [Rabiner93][Wang92]. For example, a 100 Hz tone may need to be up to 35 dB more intense than a 1000 Hz tone, for two to sound equally loud. This means that loudness of a band of noise at a constant sound pressure remains constant as the noise bandwidth increases up to the width of the critical band after that noise has been received. On the other hand, a multi-tone sound of constant intensity is about as loud as an equally intense pure tone of a frequency lying in at the center of the band, regardless of the overall frequency separation of the multiple tones. When the separation exceeds the critical bandwidth, the complex sound is perceived as becoming louder.

These perceptual non-linearities have led to modeling the peripheral auditory analysis by critical-band filters. The model postulates that sounds are preprocessed by a band of triangular filters, with center frequency spacings and bandwidths increasing with frequency. In fact, these filters are designed similarly spaced to the auditory neurons locations on the basilar membrane in the inner ear. The modified spectrum of $S(\omega)$ thus consists of the output power of these filters when $S(\omega)$ is the input. If the power coefficients is denoted by \tilde{S}_k , where $k=1,2,\dots,K$, we can calculate what is called the mel-frequency cepstrum [Davis80] denoted by \tilde{c}_n , which can be expressed as

$$\tilde{c}_n = \sum_{k=1}^K \log(\tilde{S}_k) \cos\left[n\left(k - \frac{1}{2}\right)\frac{\pi}{K}\right], \quad n = 1, 2, \dots, L$$

where L is the desired length of the cepstrum.

3.1.9. Adaptive Component Weighted Cepstrum

In this subsection a newly proposed LP-based spectral feature, called the Adaptive Component Weighting (ACW) [Assaleh94], is described. Consider the LP transfer function $H(z)$ as parameterized by the residues r_k and the poles z_k which are in turn further described by σ_k and ω_k .

The transfer function $H(z)$ can be rewritten as:

$$H(z) = \frac{1}{A(z)} = \prod_{i=1}^p \frac{1}{1 - z_i z^{-1}} = \sum_{i=1}^p \frac{r_i}{1 - z_i z^{-1}}, \quad (1)$$

where r_i are the residues and $z_i z^{-1}$ are the poles of $H(z)$. The poles are expressed as:

$$z_i = \sigma_i e^{j\omega_i}, \quad i = 1, 2, \dots, p,$$

where ω_i corresponds to the i -th center frequency. The magnitude of the poles is denoted σ_i , which falls into the range (0, 1). The bandwidth of the i -th pole is defined as:

$$B_i = \frac{1}{\pi} \ln\left(\frac{1}{|z_i|}\right) = \frac{1}{\pi} \ln\left(\frac{1}{\sigma_i}\right). \quad (2)$$

Thus, the vocal tract model corresponds to the causal impulse response given by

$$h(n) = \sum_{i=1}^p r_i z_i^n = \sum_{i=1}^p r_i \sigma_i^n e^{j\omega_i n} \quad (3)$$

The speech signal $s(n)$ is a multicomponent signal expressed as a linear combination of amplitude and phase modulated exponentials which are specified by the autoregressive model. We can write the impulse response $h(n)$ as:

$$\begin{pmatrix} h(1) \\ h(2) \\ \vdots \\ h(p) \end{pmatrix} = \begin{pmatrix} \sigma_1 e^{j\omega_1} & \sigma_2 e^{j\omega_2} & \dots & \sigma_p e^{j\omega_p} \\ \sigma_1^2 e^{j2\omega_1} & \sigma_2^2 e^{j2\omega_2} & \dots & \sigma_p^2 e^{j2\omega_p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_1^p e^{jp\omega_1} & \sigma_2^p e^{jp\omega_2} & \dots & \sigma_p^p e^{jp\omega_p} \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_p \end{pmatrix} \quad (4)$$

Formants of the speech signal are weighted by the residues r_i individually. It was observed in [Assaleh94] that the residues show considerable variation when speech is passed through a channel. This is equivalent to saying that the amplitudes r_k of the individual eigenmodes in the modulation model representation (see Eq. 3) are most perturbed by the channel among the three parameters r_k , σ_k , and ω_k . The ACW cepstrum removes the variations due to channel variability by normalizing the residues r_i so that the narrow-band components corresponding to formants are emphasized and the broad-band components are suppressed. Hence, we get a pole-zero system function of the form:

$$H_{acw}(z) = \frac{N(z)}{A(z)} = \sum_{k=1}^p \frac{1}{1 - z_k z^{-1}},$$

where

$$N(z) = \sum_{k=1}^{p-1} \prod_{i=1, i \neq k}^p (1 - z_i z^{-1}),$$

which can be further written as:

$$N(z) = p(1 - \sum_{k=1}^{p-1} b_k z^{-k}).$$

It can be shown that $N(z)$ is minimum phase [Zilovic97]. Therefore, the ACW cepstrum is causal and given by:

$$\begin{aligned} c_{acw}(0) &= \log p \\ c_{acw}(n) &= c_{lp}(n) - c_{nn}(n), \quad n > 0, \end{aligned}$$

where $c_{nn}(n)$ can be found by a recursion involving the coefficients b_k . Moreover, since b_k is simply expressed as [Zilovic97]:

$$b_k = \frac{p-k}{p} a_k, \quad 1 \leq k \leq p-1$$

the computation of the ACW cepstrum is very simple. The subtractive component $c_{nn}(n)$ serves as an estimate of the channel. In practice, rectangular weighting is applied so that the feature vector consists of the components of $c_{acw}(n)$ for $n=1$ to p . The performances of ACW cepstrum and LP cepstrum in speaker identification systems are compared in [Assaleh94][Zilovic].

3.1.10. Postfilter Cepstrum

The concept of a postfilter was introduced in [Ramammothly88] to enhance noisy speech. The philosophy in developing a postfilter relies on the fact that more noise can be perceptually tolerated in the spectral valleys (spectral peaks) than in the spectral valleys. The postfilter is obtained from $A(z)$ and its transfer function is given by:

$$H_{pfl}(z) = \frac{A(z/\beta)}{A(z/\alpha)}, \quad 0 < \beta < \alpha \leq 1$$

If $A(z)$ is minimum phase, $H_{pfl}(z)$ is guaranteed to be minimum phase. Therefore, the postfilter cepstrum (referred to as the PFL cepstrum) [Zilovic] is causal and given by:

$$c_{pfl}(0) = 0$$
$$c_{pfl}(n) = c_{lp}(n)[\alpha^n - \beta^n], \quad n > 0$$

The PFL cepstrum is merely a weighting or liftering of the LP cepstrum and is very robust to channel and noise effects [Zilovic]. Like other ways of liftering, the LP cepstrum, namely, bandpass liftering [Juang87], quefrency liftering [Paliwal82] and inverse variance liftering [Tohkura87], the lower indexed cepstral coefficients, are deemphasized. If $\alpha^n = 1$, then $c_{pfl}(n) = c_{lp}(n) - \beta^n c_{lp}(n)$. There is a subtractive component that serves as a channel estimate and which is adaptive on a frame by frame basis. In practice, rectangular weighting is applied so that the feature vector consists of the components of $c_{pfl}(n)$ for $n=1$ to p .

3.1.11. Feature Fusion

Combination of features or feature fusion (or data fusion) where one feature compliments the other can add to the overall system's performance. Data fusion method is a very useful technique to improve performance through the incorporation of different sources of information. In linear fusion, the combining logic is specified by:

$$\alpha O_1 + (1-\alpha)O_2,$$

where O_1 and O_2 are the opinions of the expert/classifier regarding the feature vectors a and b . The opinions O_1 and O_2 may be a distance measure or a probability measure.

3.2. Experimental Setup

3.2.1. Speech Database

The training and testing data for the experiments were taken from the TIMIT acoustic-phonetic continuous speech corpus². TIMIT database was created by the joint effort of Massachusetts Institute of Technology (MIT), Stanford Research Institute (SRI) and Texas Instruments (TI) for an evaluation of automatic speech recognition systems.

The database contains sentences spoken by male and female speakers in different dialects of American English language. It was designed to include dialect sentences and phonetically-compact and phonetically-diverse sentences. The dialect sentences expose the dialect differences of the speakers. The phonetically-compact group of sentences provides with the wide coverage of various pairs of phones. Phonetically-diverse sentences were selected from the Brown corpus (Kuchera and Francis, 1967) to add a variety of phonetic contexts to the first group.

New England dialect group ('dr1') utterances were used in the current research. First six speakers' (3 males, 3 females) utterances were used for training purposes, and utterances of other six speakers numbered eleven to sixteen were used in the testing routines. Two dialect ('sa') and three phonetically-diverse ('si') sentences of each speaker were used for training, and five phonetically-compact ('sx') sentences of each speaker were taken for testing. The number of male and female utterances was the same. The speech files was down-sampled from 16 KHz to 8 KHz to simulate telephone bandwidth conditions. The down-sampling was performed using a linear phase, 201 points, hamming window-design FIR filter.

Co-channel speech utterances were created by mixing two single speaker files sample-by-sample in equal proportion ($s_{mix}(n) = 0.5 * s_1(n) + 0.5 * s_2(n)$). All possible speaker combinations using all five sentences were mixed together to yield same-gender mixed as well as cross-gender mixed co-channel speech.

3.2.2. Baseline Experimental Setup

The original TIMIT 16 kHz, 16 bit linear binary utterances were downsampled to 8kHz before further processing. Table 3.2.2 below provides the setup parameters.

Parameter	Sampling Rate kHz	Frame Size msec	Frame Overlap msec	LP Order	File Mixing Weight	Gender	Number of Simultaneous Speakers
variable	8	240	160	12	0.5	Mixed	2

Table 3.2.2. Set of parameters in the baseline setup.

² TIMIT speech corpus can be obtained from Linguistic Data Consortium, University of Pennsylvania, Philadelphia, USA, (215)-898-0464, Fax: (215)-573-2175, URL: <http://www ldc.upenn.edu>

The above parameters applied together with LP cepstrum feature and vector quantizer (VQ) classifier (codebook sizes 1 through 256) algorithms create baseline experimental settings. The set of speakers used for training and testing did not overlap, i.e., to say, "open set" operating conditions.

In the successive tests one or more of the baseline conditions (the number of speakers mixed for multispeaker speech signal; the proportion in which the speech signals are mixed; the total number of speakers, etc.) are changed to investigate a particular concept. Changes made in each experiment are discussed separately.

3.2.3. Performance Metric and Decision Error Classification

The performance of the speaker count determination system is measured in terms of its accuracy in estimating the number of simultaneous talkers (one or multiple, and not actual count of talkers) in a given speech utterance on a short-time processing frame-by-frame basis.

$$\text{Performance} = \text{number of correctly classified frames} / \text{total number of frames} * 100 \%$$

The frame-by-frame errors in the system decisions are further categorized into three types as follows:

Type 1 error: single speaker speech classified as multi-speaker speech;

Type 2 error: multi-speaker speech classified as single speaker speech;

Type 3 error: misclassification between the presence and absence of speech, i.e., speech-silence misclassification.

$$\text{Error of Type } i = \text{number of frames with error of type } i / \text{total number of frames} * 100\%$$

As is evident by the above definitions,

$$\text{total number of frames} = \text{Correct} + \text{Error_T1} + \text{Error_T2} + \text{Error_T3 frames}$$

3.3. Experimental Results

The LP and cepstrum based features described above were experimentally tested with different classifiers. The "optimal" system settings were chosen as the "baseline conditions". Different variations to the baseline system and system modules were also evaluated. The following parameters and conditions have been analyzed:

1. Comparing Spectral Features Performance Using Different Classifiers
2. Adding Voicing Detection to Spectral Feature Algorithms
3. Post-processing of Final Decision Labels Using Smoothing
4. Utterance level versus Frame based Performance
5. Transitional Frames Error Analysis
6. System Performance with Non-Silence Speech

7. System Performance at Various Signal-to-Interference Ratios (SIR)
8. System Performance on an Increased Number of Speaker Population
9. Correlation between System Decision and Frame Energy
10. System Performance on an Increased Number of Simultaneously Talking Speakers
11. Feature Fusion
12. Comparing Performance of the Speaker Count System on Different Speaker Gender Combinations
13. Dependence of System Performance on Processing Frame Size
14. New Approach for Speech-Silence Discrimination Error
15. New Silence Detection Algorithm

3.3.1. Comparing the Features Performance with Different Classifiers

3.3.1.1. System Performance with Vector Quantizer (VQ) Classifier

Training data was produced for one-speaker and two-speaker cases. Each of these groups of feature vectors was used to generate a corresponding vector quantizer codebook. The codebook that renders the smallest distance to the testing feature vector identifies the speech condition. The performance was calculated based on the L_2 norm distance measure, which was performed on vector quantizers of size 1 up to 256. The results are summarized in Table 3.3.1.1.

Cdbk size	LPCC %	Area %	LAR %	LP Cep %	Q-Cep %	BPL Cep %	LSF %	Refl %	MFCC %	ACW Cep %	PF* Cep %
1	46	58	54	54	54	55	54	55	-	-	-
2	55	54	54	59	52	52	59	56	-	-	-
4	54	52	60	61	52	58	59	57	-	-	-
8	56	51	56	59	55	57	58	55	-	-	-
16	56	56	57	58	52	57	59	56	57	54	56
32	58	55	58	58	54	57	58	57	57	55	56
64	56	55	58	57	55	57	56	58	57	56	56
128	56	58	59	57	55	59	57	58	57	56	57
256	56	57	59	58	56	58	57	58	57	57	57

Table 3.3.1.1. Performance of the spectral features using a vector quantizer classifier with codebook sizes 1 to 256.

* The last three feature types (MFCC, ACW Cep, PF Cep) were tested with higher codebook sizes only.

3.3.1.2. System Performance with Neural Tree Network (NTN) Classifier

Experimental Setup

NTN classifier was used as the pattern classifier. A NTN was trained using single-speaker and multi-speaker feature data. The rest of parameters were as specified in Table 3.2.2, section “Baseline Experimental Setup”. The experiment was performed with the Linear Prediction Coding Coefficient (LPCC) and Mel Frequency Cepstrum Coefficient (MFCC) features.

Results

The results are summarized in Table 3.3.1.2.

Number of levels in tree	LPCC, %	MFCC, %
2	56	49
4	54	60
6	53	56
8	52	48
10	48	51

Table 3.3.1.2. Performance for the LPCC and MFCC features using the Neural Tree Network classifier.

Observations

The results yield to those obtained under similar conditions and utilizing VQ classifier.

3.3.1.3. System Performance with Gaussian Mixture Model (GMM) Classifier

GMM classifier works similar to VQ classifier, but also provides a cluster variance information for each cluster of data. The goal of this experiment was to investigate if supplying these extra data will improve system performance.

Experimental Setup

GMM classifier was used with the LP cepstrum feature, and number of Gaussian mixtures from 1 to 128. The rest of parameters were kept the same as in Table 3.2.2, section “Baseline Experimental Setup”.

Results

The results obtained were unpredictably low. The best score of 58% was obtained for GMM with 1 mixture component, and the scores for higher mixture

components decreased to about 30-35%. In the examination of the reasons for such a low performance, we found out that a cluster variance of multispeaker speech data was significantly smaller than a cluster variance of single-speaker data. Due to the smaller variance, the GMM probability scores for multiple speaker model were much smaller than for one speaker model. This forced to identify the majority of frames as a single speaker frame, since the GMM probability score for one speaker model was always much higher than that for multiple speaker model.

Cdbk size	LP Cep %
1	58
2	36
4	31
8	29
16	34
32	35
64	35
128	30

Table 3.3.1.3. LP cepstrum feature performance using GMM classifier.

Observations

All feasible LPC, Mel cepstrum, and cepstrum features were tested to research the problem of speaker count determination. LP cepstrum feature slightly outperformed the rest of the features. Most of the further experiments will be conducted using LP cepstrum feature and VQ classifier, which showed the better performance.

3.3.2. Adding Voicing Detection to Spectral Feature Algorithms

This experiment was conducted in order to analyze the effect of voiced frames selection on the system performance. This will allow direct comparison of spectral and pitch-based features (discussed in next chapter), since the pitch-based features are calculated using voiced frames only.

Experimental Setup

Voicing detection procedures were added to the training and testing phases. Only those speech frames, which were classified as “voiced” by the voicing detection module, were used during the training and testing. All other parameters were kept the same as in Table 3.2.2, section “Baseline Experimental Setup”.

Results

Based on the nature of voiced frames to be more “detectable and recognizable” we hypothesized that an error of type 2 – misclassification of 2-speaker frame as 1-speaker frame – should be decreased, and therefore, the overall performance of the system will be improved. However, based on the results obtained, performance of the system utilizing the voiced frames only, was a little, but consistently worse. The results for original (unvoiced + voiced) speech versus voiced only speech are summarized in Table 3.3.2 -1.

Cdbk size	LPCC O/V %	Area O/V %	LAR O/V %	LP Cep O/V %	Q Cep O/V %	BPL Cep O/V %	LSF O/V %	Refl O/V %
1	46/56	58/45	54/52	54/56	54/55	55/56	54/55	55/52
2	55/55	54/42	54/52	59/55	52/53	52/53	59/54	56/51
4	54/53	52/43	60/54	61/50	52/53	58/53	59/51	57/51
8	56/51	51/45	56/52	59/53	55/53	57/52	58/53	55/53
16	56/52	56/45	57/52	58/56	52/53	57/51	59/54	56/54
32	58/54	55/49	58/54	58/54	54/55	57/54	58/54	57/54
64	56/54	55/51	58/54	57/54	55/54	57/56	56/55	58/54
128	56/53	58/51	59/56	57/57	55/55	59/54	57/56	58/54
256	56/55	58/52	58/55	58/56	56/55	58/56	57/56	58/54

Table 3.3.2-1. Spectral features performance for original (O) and voiced only (V) speech.

Although the type 2 error indeed dropped as predicted, the type 1 error -- misclassification of 1-speaker frame as 2-speaker frame – largely increased. Tables 3.3.2-2 and 3.3.2-3 give the Err1/Err2 breakdown of original (voiced + unvoiced) and voiced only based procedures.

Cdbk size	LPCC E1/E2 %	Area E1/E2 %	LAR E1/E2 %	LP Cep E1/E2 %	Q Cep E1/E2 %	BPL Cep E1/E2 %	LSF E1/E2 %	Refl E1/E2 %
1	16/39	35/7	21/25	20/26	19/27	20/26	21/25	23/23
2	19/26	32/13	21/24	16/24	21/26	16/32	17/23	21/22
4	22/24	32/15	17/22	15/25	21/27	16/25	17/24	19/24
8	21/23	33/16	19/25	15/26	21/24	16/27	14/27	18/26
16	20/24	23/21	20/23	14/28	19/28	18/25	17/24	18/25
32	18/24	25/20	19/23	15/27	20/26	19/24	18/24	18/25
64	20/24	21/24	19/22	17/26	19/25	17/26	18/25	19/23
128	20/24	21/21	19/22	17/26	19/26	17/24	17/26	18/24
256	20/24	22/21	19/22	17/25	18/25	18/25	18/25	18/23

Table 3.3.2-2 Percentage of type1/type2 errors for original speech.

Cdbk size	LPCC E1/E2 %	Area E1/E2 %	LAR E1/E2 %	LP Cep E1/E2 %	Q Cep E1/E2 %	BPL Cep E1/E2 %	LSF E1/E2 %	Ref1 E1/E2 %
1	18/25	39/15	34/14	28/16	28/17	28/15	29/16	35/12
2	24/20	47/11	32/15	27/17	28/19	26/20	29/16	34/14
4	28/19	47/9	28/16	32/17	27/20	28/19	31/18	31/17
8	32/17	46/9	33/15	29/18	28/19	32/16	29/17	30/16
16	29/18	44/10	33/14	26/18	27/20	35/13	28/17	30/15
32	28/18	40/11	30/15	27/18	26/19	29/17	28/17	30/16
64	28/18	36/12	30/16	27/18	27/18	27/17	27/17	30/15
128	28/17	34/14	27/16	25/17	26/18	28/17	26/18	30/15
256	27/18	33/15	30/15	25/18	26/18	26/17	27/17	30/16

Table 3.3.2-3 Percentage of type1/type2 errors for voiced frames only.

Observations

As a result of voicing detection, system performance has dropped. Although, voicing detection did not improve the overall performance of the system for cepstral features, it did noticeably improve upon the type 2 error (i.e. less frames of 2-speaker frames were classified as 1-speaker frame). Therefore, voicing detection should be conducted for applications where type 2 error is more costly, and detection of multiple speakers is more important.

3.3.3. Post-processing of Final Decision Labels Using Smoothing

The final decision labels can exhibit some impulsive behavior (sporadic errors). To eliminate these sporadic errors, median filtering of the final decision has been performed. Graph 3.3.3 below shows the results of applying median filtering. On the graph, the reference decision values (solid lines) are plotted versus system final decisions (dashed lines). The top panel of the graph shows the original values of the final decision, and the bottom panel shows the adjusted values of the final decision after the median filtering. The median filtering window size is five frames. The forty frames shown are taken randomly for illustration purposes.

Results

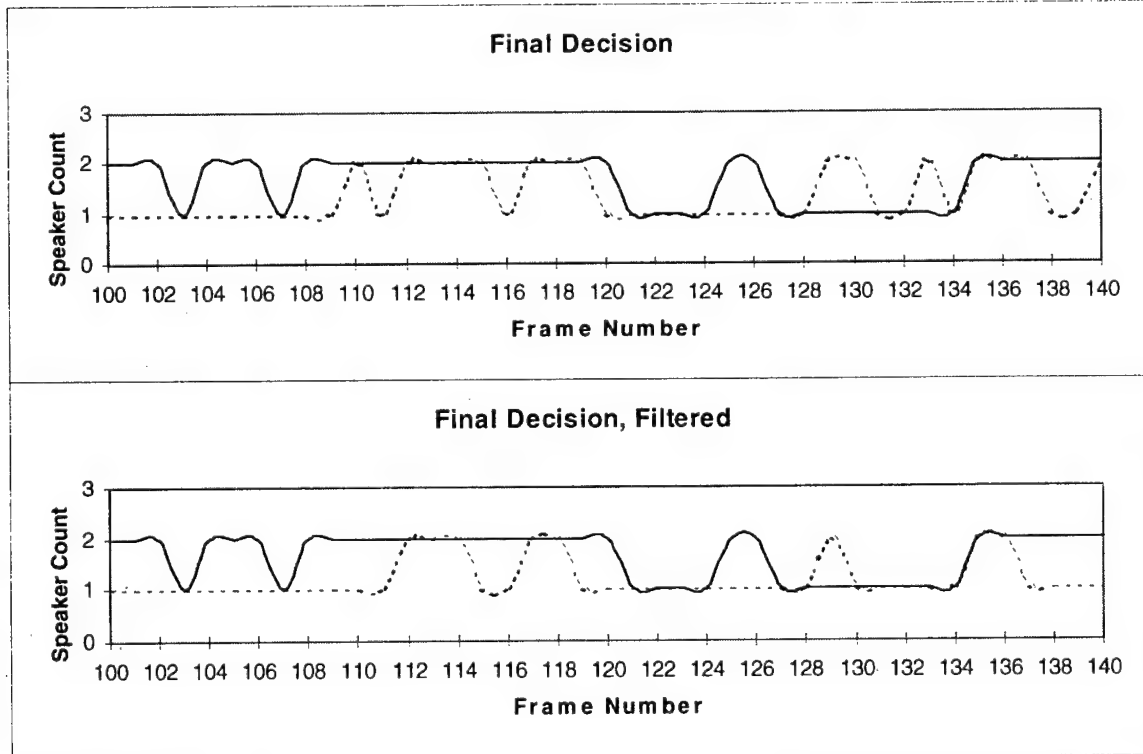
The test was performed using the baseline setup parameters described in Table 3.3. Median filtering of the final decision did not improve the system performance noticeably, as can be seen in Table 3.3.3.

Cdbk size	Final Decision Original, %	Final Decision Filtered, %
16	58	59
32	58	59
64	57	58
128	57	58
256	58	60

Table 3.3.3. System final decision: original and after median filtering.

Observations

The failure of median filtering to significantly improve the final decision can be explained by the following fact. The number of positive changes (error corrections) in the decision behavior is neutralized by the number of negative changes (error introductions). As seen in the illustration (Graph 3.3.3), in some cases there are brief periods of accurate decision in a longer "islands" of erroneous decisions. These right decisions got smoothed and became errors after filtering. This means that median filtering works successfully only if the original decision is performed accurately enough.



Graph 3.3.3. Top panel: correct decision (solid line) versus final decision (dashed line); Bottom panel: correct decision (solid line) versus final decision *filtered* (dashed line).

3.3.4. Utterance Level versus Frame Based Performance

In this test the entire utterance decision was made to estimate system performance when utterance-level decision is made. The whole utterance is labeled as single- or multispeaker, based on the label of the majority of frames composing the utterance. The parameters were set as specified in Table 3.2.2, section "Baseline Experimental Setup".

Results

Table 3.3.4 provides results for utterance level performance and corresponding performance calculated on frame by frame basis.

Cdbk size	Frame-based %	Utterance-based %
1	54	51
2	59	59
4	61	59
8	59	58
16	58	51
32	58	53
64	57	60
128	57	62
256	58	62

Table 3.3.4. Utterance-based system performance and corresponding performance calculated on frame by frame basis.

Observations

System's utterance-level performance (column three) is similar to those obtained for frame-based performance (column two).

3.3.5. Transitional Frames Error Analysis

The current experiment was performed to investigate system errors more thoroughly and to find their dependence on transitional states of speech, i.e., where the number of speakers changes. The silent segments of speech were removed from processing, leaving the single-speaker speech and multispeaker speech for the analysis. The logical assumption here was that the transitional frames were the sites, where the decision on number of speakers present could be corrupted. The goal of current experiment was to examine the number of errors occurred at transitional versus non-transitional frames.

The parameters were set as specified in Table 3.2.2, section "Baseline Experimental Setup".

During the processing each frame was labeled as "1" for single-speaker or as "2" for two-speaker speech. Then each frame was also marked as "N" for non-transitional or as "T" for transitional frames:

```

Frame label is  $L[n] = 1$  or  $2$   ("1"- single and "2"- two-speaker frame,
                        where  $n$  is a frame number)
if (  $L[n-1] = L[n] = L[n+1]$  )
mark  $M[n] = 'N'$ 
else
mark  $M[n] = 'T'$ .

```

Results

Table 3.3.5 below gives the percentage of errors for every codebook size for non-transitional and transitional frames. The total number of processed frames, and the number of frames of each type is also specified.

Cdbk size	LP Cep %; Total frames 100722	
	Non-transitional Frames - 85394 Error, %	Transitional Frames - 15328 Error, %
1	45	50
2	39	48
4	38	48
8	39	48
16	41	48
32	41	47
64	43	48
128	42	48
256	42	48

Table 3.3.5. Percent errors for non-transitional and transitional frames.

Observations

The results obtained here show that percentage of error is in low-40% for non-transitional and in high-40% for transitional frames. The transitional frames do exhibit higher error rate as expected, but are not single-handedly skewing the error rate as feared.

3.3.6. System Performance with Non-silence Speech

The test was done to estimate the performance limits of the system using silence removed speech. Eliminating the silence frames prior to all other processing will give multi-speaker speech without transitional single-to-multi-speaker and multi-to-single-speaker "contaminated" frames. When silence removed speech files are mixed together to generate co-channel multi-speaker speech, all speech frames are multiple talkers.

Experimental Setup

All silence frames were eliminated prior to all other processing. The baseline setting parameters (Table 3.2.2 of section "Baseline Experimental Setup") were used.

An extra testing procedure was added to the algorithm by including single speaker speech utterances in test set to resemble environment of real life applications. In all previous experiments the testing procedure was performed on multi-speaker speech only (which included some frames that were single speaker).

Results

Table 3.3.6 provides the results for silence-removed tests. Regular (no single speaker test utterances) and modified testing (single speaker test utterances included) routines were performed.

Cdbk size	Removed Silence, Regular Test, %	Removed Silence, Modified Test, %
1	56	56
2	60	60
4	62	62
8	60	60
16	56	57
32	55	56
64	57	57
128	60	60
256	61	61

Table 3.3.6. Silence-removed-speech experiments. Regular and modified testing routines.

Observations

The results obtained here, are about 1% higher, on average, then those obtained in the original test setting (see section 4.1). This small increase in system performance confirms the transitional frames error analysis experiment

(see section 3.3.5). It showed that non-transitional frames constituted about 85% and transitional frames made up about 15% of the total number of frames. Percentage of errors was about 40% for non-transitional and about 50% for transitional frames. Having both types of frames in the process gives us on average $(0.85 \cdot 0.4 + 0.15 \cdot 0.5) \cdot 100\% = 41.5\%$ of errors. Eliminating the transitional frames from the processing leaves us with $100 \cdot 0.4 = 40\%$ of errors. The difference, therefore, is only 1.5% that is about what we see in the current experiment.

Both types of test procedures, i.e., including single speaker utterances in test set or not, gave similar results. This proved the validity of test setup in all previous experiments.

3.3.7. System Performance at Various Signal-to-Interference Ratio (SIR)

The experiment was conducted to investigate performance of the system at various Signal-to-Interference (SIR) ratios. In a co-channel or multi-speaker corrupted speech, one speaker is assumed as the target speaker and the other speaker(s) as jammer speakers. The ratio of average signal energy between the target and jammer speakers is called the signal-to-interference ratio.

All previous experiments were done utilizing co-channel speech recordings with 0 dB SIR. Multi-speaker recordings were obtained by mixing two single-speaker speech signals together (target and jammer speakers) with the same weights. To obtain 6 dB steps of SIR above 0 dB (i.e. less noise, or less contribution of jammer speaker), successive multiplication of a jammer speaker signal by 0.5 (decreased ratio) was performed when mixing files. To obtain SIR below 0 dB, the target speech signal has to be multiplied by 0.5 when combining files together.

Table 3.3.7 below gives the ratios α and β at which target and jammer files were combined: multi-speaker (mixed) file = $\alpha * s_t + \beta * s_j$, where

s_t is a target speaker signal,
 s_j is a jammer speaker signal.

	0 dB,	6 dB,	-6dB,	12 dB,	18 dB,	24 dB,
α	0.5	0.5	0.25	0.5	0.5	0.5
β	0.5	0.25	0.5	0.125	0.0625	0.03125

Table 3.3.7. Ratios α and β at which target and jammer signals were combined

Experimental Setup

The “file mixing weight” parameter (Table 3.2.2 of section “Baseline Experimental Setup”) was altered for each experiment, codebook sizes 16 to 256 were tested, all other baseline parameters were left unchanged.

Results

The results are shown in Table 3.3.7.

Cdbk size	0 dB, %	6 dB, %	-6dB, %	12 dB, %	18 dB, %	24 dB, %
16	58	56	56	53	50	48
32	58	56	57	53	50	49
64	56	55	55	53	52	48
128	57	57	55	53	51	49
256	58	56	56	54	51	50

Table 3.3.7. Performance at different SIR ratios for LP cepstrum feature

Observations

Since the current test was only to determine the number of speakers present in a given frame of speech, neither speaker in the multi-speaker conversation has been directly considered as “target “ or “jammer”, and the results at $\pm x$ dB were similar. The performance of the system monotonically decreased as SIR ratio changed from 0 to 24dB.

3.3.8. System Performance with Increased Number of Speaker Population

To improve statistical significance of the system results, the experiment was run on an increased number of speakers.

Experimental Setup

Thirty-nine instead of six speakers were used for training and testing. The codebook sizes 16 to 256 were tested. The rest of the setup parameters were kept the same as in baseline setting parameter (Table 3.2.2 of section “Baseline Experimental Setup”).

Speakers used for training and testing were the same, which defined the “closed-set” conditions. However, the utterances used for training and testing were different.

Results

The results obtained for 39-speaker runs are consistent with those obtained for 6 speakers, open-set runs. Table 3.3.8 below provides the results.

Cdbk size	LPCC O6/C39 %	Area O6/C39 %	LAR O6/C39 %	LPCep O6/C39 %	QCep O6/C39 %	BPLCep O6/C39 %	LSF O6/C39 %	Refl O6/C39 %
16	56/55	56/52	57/57	58/56	52/54	57/56	59/57	56/57
32	58/55	55/53	58/57	58/56	54/55	57/57	58/57	57/58
64	56/56	55/56	58/58	57/57	55/56	57/57	56/58	58/58
128	56/56	58/57	59/59	57/58	55/57	59/58	57/58	58/58
256	56/56	58/57	58/59	58/58	56/57	58/58	57/59	58/59

Table 3.3.8. Performance for original 6-speaker “open-set “ tests (O6) versus current 39-speaker “closed-set” tests (C39). LP-based cepstral features.

Observations

The results of current experiment are similar or better to the results obtained previously in the test-runs for 6-speaker, "open set" environment. This also justifies the validity of baseline experimental setup.

3.3.9. Correlation between System Decision and Frame Energy

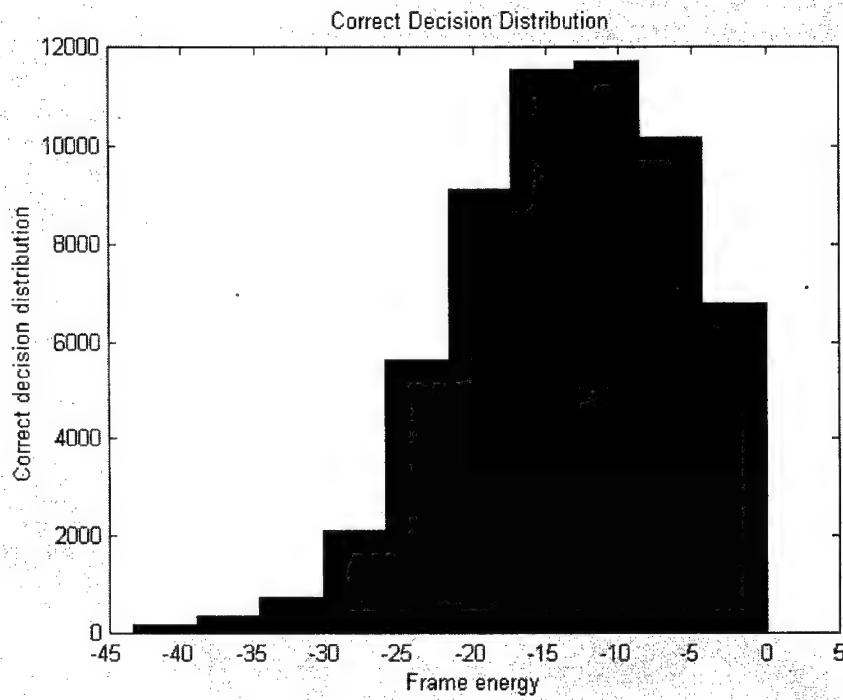
The experiment was performed to find the correlation between a frame labeling decision and the energy of that frame.

Experimental Setup

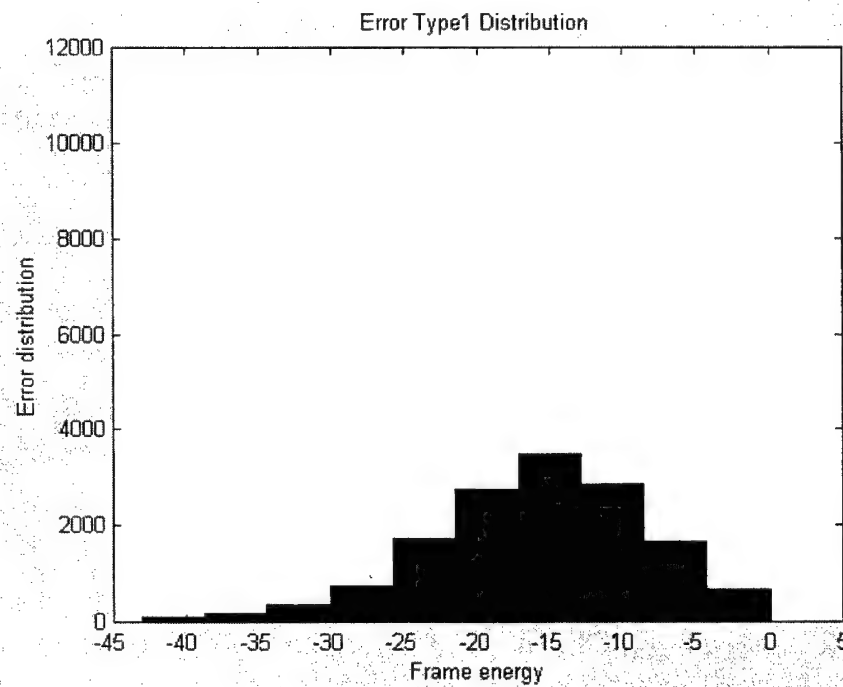
The experimental conditions in this study were the same as in the baseline setting (Table 3.2.2 of section “Baseline Experimental Setup”). For codebook of size 16, each frame’s energy value was recorded. The distribution of system decisions was analyzed as a function of the frame energy value.

Results

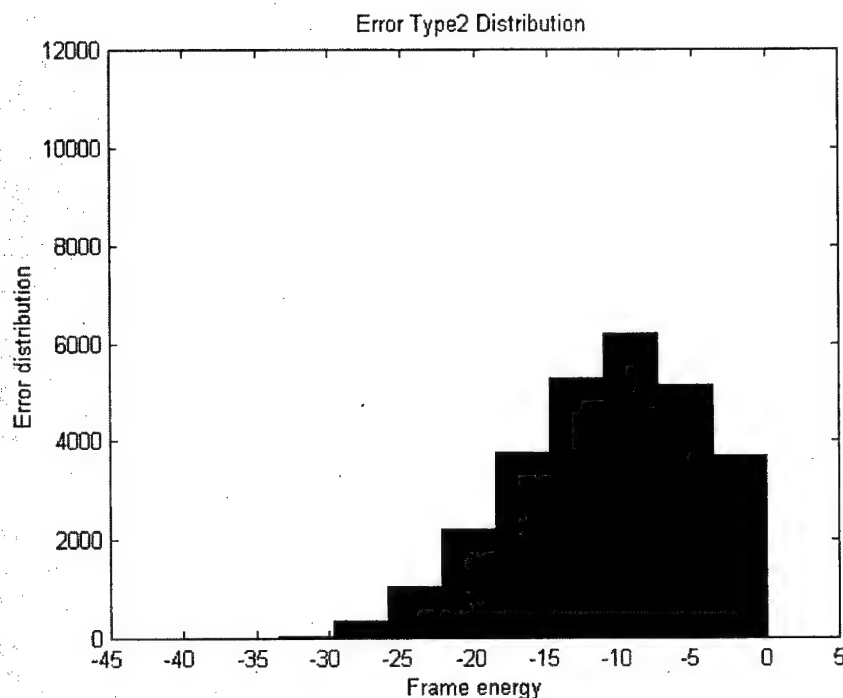
Graphs 3.3.9-1 to 3.3.9-3 below show the distribution of correct decisions and errors of type 1 and type 2 as a function of frame energy. The number of type 3 errors is very small, and errors are localized at the energy range approximately –37 dB to –27 dB. The frame energy values are normalized so the maximum value is equal to 0 dB.



Graph 3.3.9-1 Distribution of correct system decisions as a function of frame energy



Graph 3.3.9-2 Distribution of errors of type1 as a function of frame energy



Graph 3.3.9-3 Distribution of errors of type 2 as a function of frame energy

Observations

As it is seen from the graphs, system decisions - either correct or erroneous (except type 3 errors) - fall mostly at the same frame energy range -20 dB to -5 dB, which makes it difficult to design any ancillary criterion for reparability based on energy.

3.3.10. System Performance on an Increased Number of Simultaneously Talking Speakers

This test was run to test system performance on the speech where the number of speakers talking simultaneously is more than two.

Experimental Setup

The experiments were run with co-channel speech obtained by mixing utterances from three speakers, four speakers and five speakers. The rest of the setup parameters were kept the same as in baseline setting parameter (Table 3.2.2 of section "Baseline Experimental Setup").

For "3-speaker-test" five sentences of each speaker were utilized. For "4-speaker-test" three different sentences of each speaker were taken to create

multispeaker speech combinations. For “5-speaker-test” two sentences of each speaker were used.

Results

As the number of simultaneously talking speakers increased up to three, four and then five (instead of two speakers used originally) the system's performance improved, as seen in Table 3.3.10-1 below.

Codebook size	Original test	3-speaker test	4-speaker test	5-speaker test
Original Setup, LPCep, %				
1	54	60	60	65
2	59	65	66	59
4	61	65	66	70
8	59	65	66	71
16	58	63	66	69
32	58	61	64	71
64	57	61	63	69
128	57	62	64	70
256	58	62	63	68

Table 3.3.10-1. Results for a different number of speakers present in multispeaker speech; the original setup.

An investigation of system errors for “3-speaker”, “4-speaker” and “5-speaker” tests showed that error distribution is quite uneven, where Type 2 error is three times more than the error of Type 1. This fact prompted the idea of “selective” smoothing of the final system decision, i.e. to median filter out the decision errors only when the frame of speech was classified as 1-speaker speech instead of multispeaker speech.

The “selective” smoothing algorithm, using median filter, works as follows:

Decision label is $L[n] = 1$ or 2 (“1”-single and “2”- multispeaker frame;
 $n = 1, \dots, N$, where $N = 5$ (median filter window size))

if (count ($L[n] = 2$) > count ($L[n] = 1$))

then median filter

else

continue.

The results of applying “selective” median filtering are quoted in Table 3.3.10-2.

Codebook size	2-speaker test	3-speaker test	4-speaker test	5-speaker test
	Selective Smoothing, LPCep, %			
1	60	61	62	66
2	60	67	69	62
4	62	68	70	74
8	61	69	71	75
16	60	68	72	74
32	60	67	71	77
64	59	67	70	75
128	59	68	71	77
256	60	69	70	76

Table 3.3.10-2. Experimental results for different number of speakers present in multispeaker speech after “selective” smoothing was applied.

Observations

The results of current experiment demonstrate that the system performance improves as the number of interfering speakers increases. The error breakdown tends to be biased towards Type 2 error when the number of simultaneous speakers is increased. Because of the fact of an uneven error distribution, selective filtering of the final decision could be successfully applied. The probability of correct replacements of the frame labels mislabeled as one-speaker by multispeaker frame label grows as the number of simultaneously interfering speakers increases.

3.3.11. Feature Fusion

For the speaker count problem, the linear fusion of different features was used to calculate the distance (or probability) measures of VQ codebooks for the one-speaker and two-speaker cases.

3.3.11.1. Fusion of the Features with Complementary Error Distributions

Two features with complementary error distributions were chosen – LP cepstrum and Area Coefficients. The rest of the parameters was set as described in the “Baseline Experimental Setup” section (Table 3.2.2).

Two strategies for feature fusion have been examined here:

Features concatenation: append the features together and continue processing as for single feature scenario.

Score fusion: extract and save the features separately, obtain minimal distances with 1-speaker and 2-speaker codebooks for each feature on frame by frame basis, fuse each feature analogous 1-speaker and 2-speaker distances together,

determine the decision by finding a smaller distance.

Results

1. Features concatenation

In this study the two features vectors were concatenated together to yield a composite feature vector of double length. These vectors were processed in the same way as a single feature. This technique was implemented and tested. The results are quoted in Table 3.3.11.1-1.

Cdbk size	Area/LP Cep Features concatenation, %	Area %	LP Cep %
1	58	58	54
2	54	54	59
4	52	52	61
8	51	51	59
16	56	56	58
32	56	55	58
64	55	55	57
128	58	58	57
256	59	57	58

Table 3.3.11.1-1. Feature fusion. Combined processing of Area Coefficients and LP cepstrum features – column1. Columns 2 and 3 – results obtained for each feature used alone.

2. Score fusion

Another way to fuse the features is to process each feature separately, to find minimal distances for each feature codebooks and fuse the distances together. When codebook distances were calculated, it occurred that area-coefficients-feature distances lie in the much bigger range than distances obtained with LP cepstrum feature. Therefore, some kind of distance normalization was necessary. The normalization was performed using two different approaches. In the first approach, each frame distance was normalized to be in the range (0,1) when it was calculated. The results of this normalization are shown in the first column in Table 3.3.11.1-2. In the second approach, all frames distances were calculated first and then normalized together based on common minimum and maximum values. The results of this method are posted in the second column of Table 3.3.11.1-2.

Cdbk size	Area/ LP Cep Score Fusion, Normalize each frame distances, %	Area/ LP Cep Score Fusion, Normalize all distances together, %
1	47	57
2	49	54
4	48	52
8	49	51
16	49	56
32	50	58
64	50	57
128	49	59
256	49	59

Table 3.3.11.1-2. Feature fusion. Separate processing of LP cepstrum and area coefficients and combining the final distances. Two types of distance normalization is applied: each frame distance (column one) and all frames together (column two).

Observations

The results of the fusion utilizing a separate feature processing are almost the same as those obtained for the Area Coefficients feature alone. This means that the Area Coefficients feature dominated the LP cepstrum feature, which happened because of the difference in the codebook distance ranges. In the second experiment, where the separate feature processing and distance normalization were applied, the Area Coefficients feature still prevailed over the LP cepstrum feature. The results obtained here showed that the features chosen in the experiment could not be fused successfully. This happened because the difference in their characteristics was too big. This fact should be considered for future choices of feature combinations.

3.3.11.2. Fusion of the LP Cepstrum and Short-term Variance (delta) Features

All setup parameters set as in the baseline settings ("Baseline Experimental Setup" section Table 3.2.2). LP cepstrum and short-term variance (delta) features were calculated and used for fusion. Two strategies for feature fusion have been used here, as described in the previous feature fusion experiments: features concatenation and score fusion methods.

Results

The results are provided in Table 3.3.11.2. It can be seen that both methods of feature fusion give almost equal results, comparable with those obtained using LP cepstrum feature alone.

Cdbk size	LP Cep/Delta Score Fusion %	LP Cep/Delta Features Concatenation %	LP Cep %
1	55	53	54
2	57	59	59
4	60	54	61
8	60	60	59
16	60	58	58
32	59	59	58
64	58	58	57
128	58	57	57
256	59	58	58

Table 3.3.11.2. LP cepstrum and delta feature fusion: separate (score fusion) and combined (features concatenation) feature processing. The last column – results for LP cepstrum feature alone.

Observations

Fusion of LP cepstrum and delta features did not exhibit any significant improvement in the system performance.

3.3.12. Comparing Performance of the Speaker Count System on Different Speaker Gender Combinations

The experiment was to compare the system performance on speech from different gender speakers (male versus female).

Experimental Setup

In all the previous experiments an equal number of male (M) and female (F) speaker speech files were utilized for creating mixed or multi-speaker speech files. Therefore, the resulting multispeaker files were of any possible combination: M + M, M + F, F + F. Here the performance of the system was studied under controlled condition of gender mixing:

1. Male speech only mixing – M + M;
2. Female speech only mixing – F + F;
3. Male speech + female speech combination – M + F;

In all three experiments the baseline setup conditions were applied (Table 3.2.2 of section “Baseline Experimental Setup”).

Results

The results are provided in Table 3.3.12 for each of the above experiments.

Cdbk size	Male speech only	Female Speech only	Male + Female Speech
1	52	53	50
2	54	56	56
4	57	59	57
8	58	59	58
16	56	58	56
32	58	59	57
64	57	59	58
128	57	59	58
256	57	59	58

Table 3.3.12. System results for different speaker gender combinations.

Observations

The results show very similar performance, with slightly better completion in the female/female speech combination test.

3.3.13. Dependence of System Performance on a Processing Frame Size

All setup parameters were set as in the baseline setting (Table 3.2.2 of section "Baseline Experimental Setup"), except for a frame size, which was altered from 80 milliseconds to 480 milliseconds, in steps of 80 milliseconds. In every frame size case, the frame overlap between adjacent frames was set to 2/3rd of the frame size.

Results

Table 3.3.13 below summarizes the results.

Cdbk size	Frame Size 80	Frame Size 160	Frame Size 240	Frame Size 320	Frame Size 400	Frame Size 480
	LP Cep, %					
1	53.9	54.2	54.3	54.3	54.8	54.7
2	59.3	59.5	59.1	58.9	58.6	57.9
4	58.1	61.5	60.5	59.3	58.6	57.9
8	58.6	59.9	59.2	59.2	59.0	58.7

16	57.1	58.6	57.9	57.8	57.9	58.7
32	57.9	57.7	57.7	56.6	55.7	57.1
64	58.1	57.6	56.5	55.9	56.3	56.4
128	57.7	57.9	56.9	56.0	56.5	56.1
256	58.2	58.0	57.5	56.9	56.8	55.6

Table 3.3.13. System performance for different frame processing size.

Observations

The results vary in the range of several percent. The frame size of 160 milliseconds was marginally the best on average.

3.3.14. New Approach for Speech-silence Discrimination Error

Speech-silence discrimination (Type 3) error calculations are revisited and adjusted here. In earlier experiments, the speech-silence discrimination (Type 3) error was accumulated only when the system decision was estimated as non-silence (i.e., single or multi-speaker) when the reference label was silence (or 0). If the estimated system decision for a frame was silence (0), it was not ignored from error computation and compared to the reference label.

In summary, the type 3 error was earlier defined as:

if (reference is "*silence*" but decision is "*non-silence*")
then type 3 error defined:

Reference label	Decision label	result
0	1 or 2	Type 3 error
1 or 2	0	Not counted

Table 3.3.14-1. Old speech-silence discrimination (Type 3) error definition

In the current experiment, the Type 3 error is redefined as follows:

If (reference is "no speech present" but decision is "speech present")
Or
(reference is "*non-silence*" but decision is "*silence*")
then type 3 error defined:

Reference label	Decision label	Result
0	1 or 2	Type 3 error
1 or 2	0	Type 3 error

Table 3.3.14-2. New speech-silence discrimination (type 3) error definition

With the new approach for Type 3 error calculation, and the baseline condition settings (Table 3.2.2 of section "Baseline Experimental Setup") the experiment gave the new results, which are provided in Table 3.3.14-3. The old results are provided for comparison.

Cdbk size	Correct *		Type 1 error *		Type 2 error *		Type 3 error	
	Old	New	Old	New	Old	New	Old	New
1	54	60	20	15	26	20	0.11	6.27
2	59	63	16	12	24	18	0.11	6.27
4	61	64	15	11	25	19	0.11	6.27
8	59	63	15	11	26	19	0.11	6.27
16	58	62	14	11	28	21	0.11	6.27
32	58	62	15	11	27	21	0.11	6.27
64	57	61	17	13	26	20	0.11	6.27
128	57	61	17	13	26	20	0.11	6.27
256	58	62	17	13	25	19	0.11	6.27

Table 3.3.14-3. System results for different Type 3 error approaches using LP cepstrum feature.

*Correct result and errors of Types 1 and 2 are rounded up, so their total may not be exactly equal to 100%.

Observations

The single and multi-speaker discrimination errors i.e., error types 1 and 2, decrease using the new definition of speech-silence discrimination error. This is due to the fact that previously those frames with estimated decision label equal to silence ("0") were not counted in the statistics; however, they are now included in the total count as per the new definition. Therefore, although the number of frames with error Type 1 or 2 remain the same as before, the total number of frames counted (in denominator) to obtain the ratio has increased.

Moreover, by including the test frames with estimated decision label as silence, the speech-silence discrimination error (Type 3) has increased, as expected. Overall, the results of experiments show about 3-4 percent of improvement when the new type 3 Error calculation algorithm was applied.

3.3.15. New Silence Detection Algorithm

The previous silence detection algorithm first computed the mean and variance of a given speech, and then determined a threshold based on the values of mean and variance. To decrease the system misclassification between speech and silence (Type 3 error), new silence detection algorithm was studied. This silence removal algorithm is based on the observation that in most speech corpus the spoken utterances are predominantly silence. The new algorithm first calculates

a histogram of frame-by-frame speech energy, then picks the peak in the histogram and uses the peak value as an energy threshold. This threshold energy value is used to perform speech versus silence discrimination. The speech frames having energy less than the threshold value, are classified as silence frames.

The "old" results quoted here are computed with the new Type 3 error calculation method.

Cdbk size	Correct * %		Type 1 error * %		Type 2 error * %		Type 3 error %	
	Silence Detection Algorithm		Silence Detection Algorithm		Silence Detection Algorithm		Silence Detection Algorithm	
	Old	New	Old	New	Old	New	Old	New
1	60	67	15	16	20	14	6.27	2.4
2	63	69	12	15	18	14	6.27	2.4
4	64	68	11	15	19	14	6.27	2.4
8	63	69	11	14	19	14	6.27	2.4
16	62	68	11	14	21	15	6.27	2.4
32	62	68	11	15	21	15	6.27	2.4
64	61	68	13	15	20	14	6.27	2.4
128	61	68	13	15	20	15	6.27	2.4
256	62	68	13	16	19	14	6.27	2.4

Table 3.3.15. System performance with a new silence detection algorithm

*Correct result and errors of Types 1 and 2 are rounded up, so their total may not be exactly equal to 100%.

Observations

The experimental results show about 6-7 percent of improvement when new silence detection algorithm was applied. The improvements can be attributed to a more robust silence detection algorithm.

4. Pitch Based Methods

The other approach to discriminate between speech containing single or multiple speakers is to use pitch based features. Voiced speech involves vibration of the vocal cords, and pitch refers to the fundamental frequency of such vibration, or the resulting periodicity in the speech signal. It is the primary acoustic cue to intonation and stress in speech, and is useful in speech coding, speech synthesis and speaker recognition. With regard to the speaker count problem, it is hypothesized that if there is only single speaker, one pitch value and its multiples will be detected. If there are more than one speaker, multiple pitch values will be detected. Two types of algorithms have been explored for speaker count determination problem: the pitch prediction [Ramachandran89] and LPC residual based pitch concepts.

4.1. Pitch Prediction Feature (PPF)

The theory of Pitch Prediction is visited here. There are two major types of correlations (or, redundancies) present in a speech signal. These correlations are known as the near sample correlation and the distant sample correlation. The near-sample correlation refers to the formant information, whereas the distant sample correlation refers to the inherent periodicity of voiced speech. Predictive speech coders make use of this correlation in the speech signal to enhance coding. In predictive speech coders, the cascade of two non-recursive prediction error filters is used on the original speech. The first filter uses linear predictive filter to remove the near-sample redundancies, whereas the pitch filter acts on the distant sample redundancies.

The result is a residual signal that has little sample to sample correlation. The parameters, that are then coded and quantized for transmission, include the filter coefficients and the residual signal. From these coded parameters, the receiver synthesizes the speech by first re-inserting the fine pitch information and then, shaping the spectral envelop to re-insert the formant information. An algorithm based on similar principles is proposed for speaker count determination.

The first filter or the formant predictor based on LPC filter has the following transfer function:

$$F(z) = \sum_{i=1}^Q a_i z^{-i},$$

where, the LPC filter order Q is between 8 and 16 for a 8 kHz sampled speech. The speech signal $s(n)$ is passed through the filter $1-F(z)$ to generate a residual signal $r(n)$ that is free of near-sample correlation. The corresponding synthesis filter is $H_F(z)=1/(1-F(z))$. The coefficients a_i can be found using the Levinson-

Durbin algorithm. This method finds the coefficients by minimizing the weighted mean-square error of $r(n)$ over a frame of N samples [Rabiner78].

The second filter is the pitch predictor filter. The simplest form of the pitch predictor filter has one tap whose transfer function is given by:

$$P(z) = \beta_1 z^{-M},$$

where, the integral delay M represents the pitch period. Since the sampling frequency is unrelated to the pitch period, the individual samples do not show a high period to period or distant sample correlation [Atal74]. Therefore, a 3 tap predictor serves as an interpolation filter and provides with interpolated estimates that show higher period-to-period correlation. The transfer function is:

$$P(z) = \beta_1 z^{-M+1} + \beta_2 z^{-M} + \beta_3 z^{-M-1}.$$

In computing the predictor coefficients β_i and M , consider the situation of a signal that is passed through the prediction error filter $1-P(z)$ to generate a residual $e(n)$. The signal can either be the input speech $s(n)$ or the residual $r(n)$ formed after LP analysis and filtering. Assuming a given value of M , the coefficients of $P(z)$ are chosen to minimize the mean-squared residual:

$$E_{mse} = \sum_{n=1}^N e^2(n),$$

$$\text{where } e(n) = r(n) - \beta_1 r(n-M+1) - \beta_2 r(n-M) - \beta_3 r(n-M-1),$$

$r(n)$ is the input signal and N is the number of samples in one frame.

The minimization of E_{mse} leads to a system of equations which can be written in matrix form as $Ac = d$. For a 3 tap predictor, the entries of the matrix A are:

$$A(i, j) = \phi(M+i, M+j) = \sum_{n=1}^N r(n-M-i)r(n-M-j), \quad -1 \leq i, j \leq 1.$$

The vector c is:

$$c = [\beta_1 \ \beta_2 \ \beta_3]^T$$

and the vector d is:

$$d = [\phi(0, M-1) \ \phi(0, M) \ \phi(0, M+1)]^T$$

Note that for the 1 tap case, the predictor coefficient is determined as

$$\beta_1 = \phi(0, M) / \phi(M, M).$$

Methods to determine M are described in [Ramachandran89]. The methods are based on analyzing an expression for the resulting minimum mean-squared error

E_{res} given that the coefficients are obtained by solving the system of equations given above. The resulting error E_{res} is:

$$E_{res} = \phi(0,0) - \mathbf{c}^T \mathbf{d}$$

in which the second term is a function of M . The optimal value of M is that which maximizes $\mathbf{c}^T \mathbf{d}$. The procedure is to do an exhaustive search of all integral values of M within an allowable range (usually between 20 and 147 samples for 8 kHz sampled speech) to find the optimal value. For the 1 tap case, the expression $\phi^2(0,M) / \phi(M,M)$ is maximized. However, for 3 tap predictors, it is computationally expensive to do an exhaustive search. Two sub-optimal approaches as outlined in [Ramachandran89] are as follows. First, the value found for the 1 tap case can be used for 3 tap filters. The second approach assumes that the input signal is $r(n)$. In this case, the off-diagonal terms of \mathbf{A} which represent near-sample redundancies can be neglected. Then, an approximation to $\mathbf{c}^T \mathbf{d}$, which facilitates an exhaustive search, is given by:

$$\mathbf{c}^T \mathbf{d} \approx \sum_{m=M-1}^{M+1} \frac{\phi^2(0,m)}{\phi(m,m)}$$

The value of M that maximizes $\mathbf{c}^T \mathbf{d}$ is the estimated pitch period of the primary speaker. However, other local maxima will indicate multiples of the pitch period of the primary speaker and the possible presence of other pitch values due to other speakers. All local maxima of $\mathbf{c}^T \mathbf{d}$ are found. The differences between the successive local maxima are computed. The standard deviation of these differences is the feature. For example, if there is one speaker with a pitch period of 40 samples, the local maxima will occur at 40, 80 and 120. The values of the difference in the maxima are all 40 and the standard deviation is 0. In practice, the standard deviation is low when there is only one speaker. If there are two speakers with pitch periods of 40 and 50, the local maxima will be at 40, 50, 80, 100 and 120. The values of the difference are 40, 10, 30, 20 and 20. The standard deviation is 11.4. In practice, the standard deviation is high when there are two speakers.

The pitch prediction feature (PPF), therefore, is defined as the standard deviation of the difference between the local peaks of the quantity $\mathbf{c}^T \mathbf{d}$ as determined by the pitch prediction method, where $\mathbf{c}^T \mathbf{d}$ is derived from a pitch period. The local peaks are those peaks of $\mathbf{c}^T \mathbf{d}$ that are above a given threshold. Based on observations, peaks that are greater than 50% of the global maximum have been chosen as possible pitch peaks. If a frame of co-channel speech signal has one speaker, strong peaks will occur at multiples of the pitch period. Therefore, the standard deviation of the differences of the peaks will be very small. When there are two or more speakers in a speech segment there will be a considerable larger number of strong peaks for a set threshold. This is due to the strong cross-correlation values between the pitch periods of the two or more speakers in the frame. For this reason, the standard deviation of the differences of the peaks will be much higher. If there is unvoiced speech or silence in the

frame the silence and voiced speech detector is used to preprocess the speech so that these frames are identified and not considered at this point for speaker count determination.

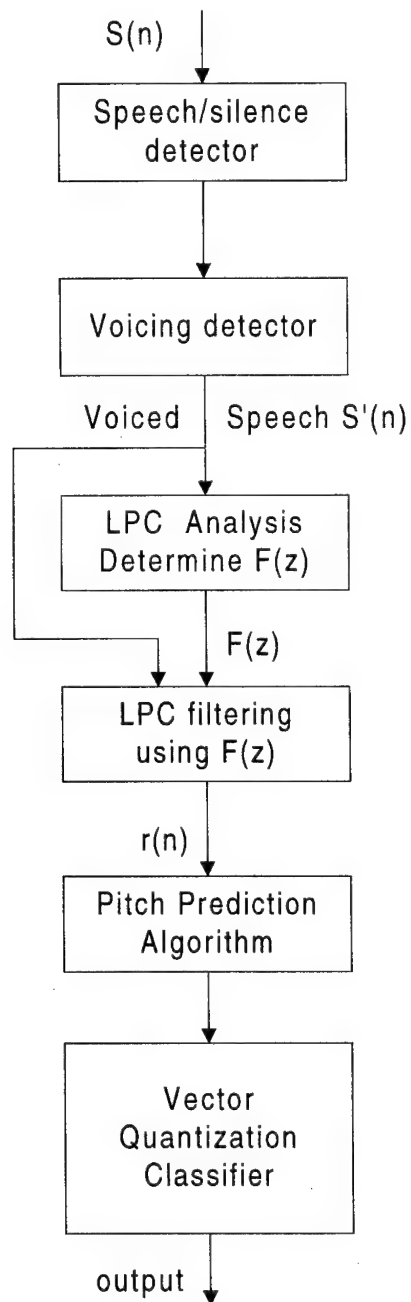


Figure 4.1. Pitch Prediction Algorithm for Speaker Count Determination

The pitch prediction feature algorithm is applied to 10 millisecond analysis frames (80 samples) of voiced speech, or what is perceived to be voiced speech signals. Twelfth order linear prediction analysis ($Q=12$) is first applied to the

signal to determine the linear prediction coefficients. The speech signal is then filtered to remove the formant information making the peaks more pronounced so that the pitch prediction algorithm can perform much better in determining the pitch peaks. The output of the pitch prediction algorithm is then fed into the vector quantizer classifier. An illustration of the training phase of the algorithm is shown in Figure 4.1 above.

A step by step PPF algorithm of the training and testing phases for speaker count determination is summarized below.

The Training Algorithm

Speech from one speaker only:

- Energy thresholding is performed to ignore frames of silence. A simple voice detector is utilized to only consider voiced frames. We are only looking at voiced frames since the problem of speaker count for unvoiced frames is a very difficult that will be dealt with further in the future.
- Twelfth order LP analysis and LP filtering are performed on the speech signal to remove the formant information. This improves the accuracy of the pitch prediction algorithm.
- Find PPF based on LP residual for each frame.

Speech from two speakers:

- Perform energy thresholding and voicing detection for the individual speech frames of each speaker.
- Determine which frames have one speaker or two speakers.
- Find PPF based on linear prediction filtered residual speech for each frame.
- Generate training data for one-speaker frames and two speaker frames from above two steps.
- From this data, design vector quantizer classifier by LBG method for each case.

The Testing Algorithm

- Mix normalized speech signals of two different speakers.
- Perform energy thresholding and voicing detection. This is to consider voiced frames only.
- For each voiced frame:
 - Find PPF based on LP residual speech
 - Compute distances to each codebook (single speaker and multiple speakers)

x_1 = smallest distance to codebook for one-speaker case

x_2 = smallest distance to codebook for two-speaker case

If $x_1 \leq x_2$, one speaker is present

If $x_1 > x_2$, two speakers are present.

4.2. LPC Residual Based Pitch Feature

In this algorithm, variance of pitch frequency estimated within a frame, was obtained using LPC residual signal. In an LPC model, the vocal tract is represented by an all-pole filter $H(z)$:

$$H(z) = \frac{1}{1 - A(z)}.$$

The coefficients of $A(z)$ are obtained through a linear predictive analysis of the speech signal. The linear predictor, $A(z)$, generates the estimate of the speech signal from the input speech signal. This estimate is then subtracted from the speech signal to obtain the error signal, $e(n)$, which is called the prediction residual signal. The error signal, is generated by the inverse filter, given by:

$$\frac{1}{H(z)} = 1 - A(z)$$

The predictor coefficients are estimated by minimizing the energy of the prediction residual, E , given by:

$$E = \sum_n e^2(n)$$

with respect to the prediction coefficients. In the expression, $e(n)$ is the output of the inverse filter and is given by:

$$e(n) = s(n) - \sum_{i=1}^p a_i s(n-i)$$

For voiced speech, an error signal is a train of pseudo-periodic pulses, while for unvoiced sound it is a white noise signal. The distance between the consecutive signal pulses gives the pitch period value.

It is hypothesized that in a single-speaker speech, the voiced LP residual (the error) signal should be a train of pseudo-periodic pulses. If multiple speakers are present in the speech, the systematic structure no longer exists. Therefore, the distance between the locations of the peaks in the LP residual signal should be about the same for single-speaker and quite erratic for multi-speaker speech.

Variance of the peak distances can become a good measure to determine the speaker number and serve as a discriminative feature for the Speaker Count Determination problem.

An effective pitch based speaker count determination system would account for commonly found pitch estimation errors, such as pitch doubling and pitch halving. These errors result from misidentifying the fundamental frequency (pitch) and its first harmonic. Pitch doubling tends to occur when the energy level in the fundamental is weak compared with adjacent harmonics. Pitch halving occurs when two pitch periods are mistaken for one. In the proposed approach, the pitch doubling and pitch halving effects are accounted for grouping the peak distances into two clusters (one cluster for actual pitch values, and another for double or half the pitch value, if applicable).

Figure 4.2. on the next page shows a scheme of Speaker Count algorithm using LPC residual based pitch feature. No separate training module is required for this method, since the proposed algorithm is rule-based. The combined intra-cluster variance of estimated pitch values is compared against a pre-determined threshold value for final decision.

LPC residual based pitch approach was thoroughly analyzed using signal processing and visualization tools available in MATLAB. Plots from these visualization studies are discussed in the following sections.

The silence frames in speech data were removed before mixing the signals, so all frames in a multi-talker speech file were co-channel speech. LPC residuals of the speech were analyzed. The speech analysis frame size in the experiment was 30 milliseconds. The LPC analysis order was 12. Frames of speech in multi-speaker files were marked as "1" for 1-speaker speech frame, and as "2" for 2-speaker speech frame.

Various scenarios of single and multiple talker speech signal composition were investigated:

1. Typical case,
2. Pitch doubling,
3. Pitch halving,
4. Female speaker dominating,
5. Male speaker dominating.

The following figures show the analysis of speech residual signal. In each figure the subplots represent the following:

First pane (subplot) show the entire original speech signal, with the specific frame location under investigation marked.

Second pane shows the details of the speech signal of one particular frame selected for analysis.

Third pane shows the speech residual signal, where the 'x' marks the location of the automatically located peaks.

Bottom left pane shows the histogram of the distances between the located peaks.

Bottom right pane shows the LP spectrum of the corresponding speech frame.

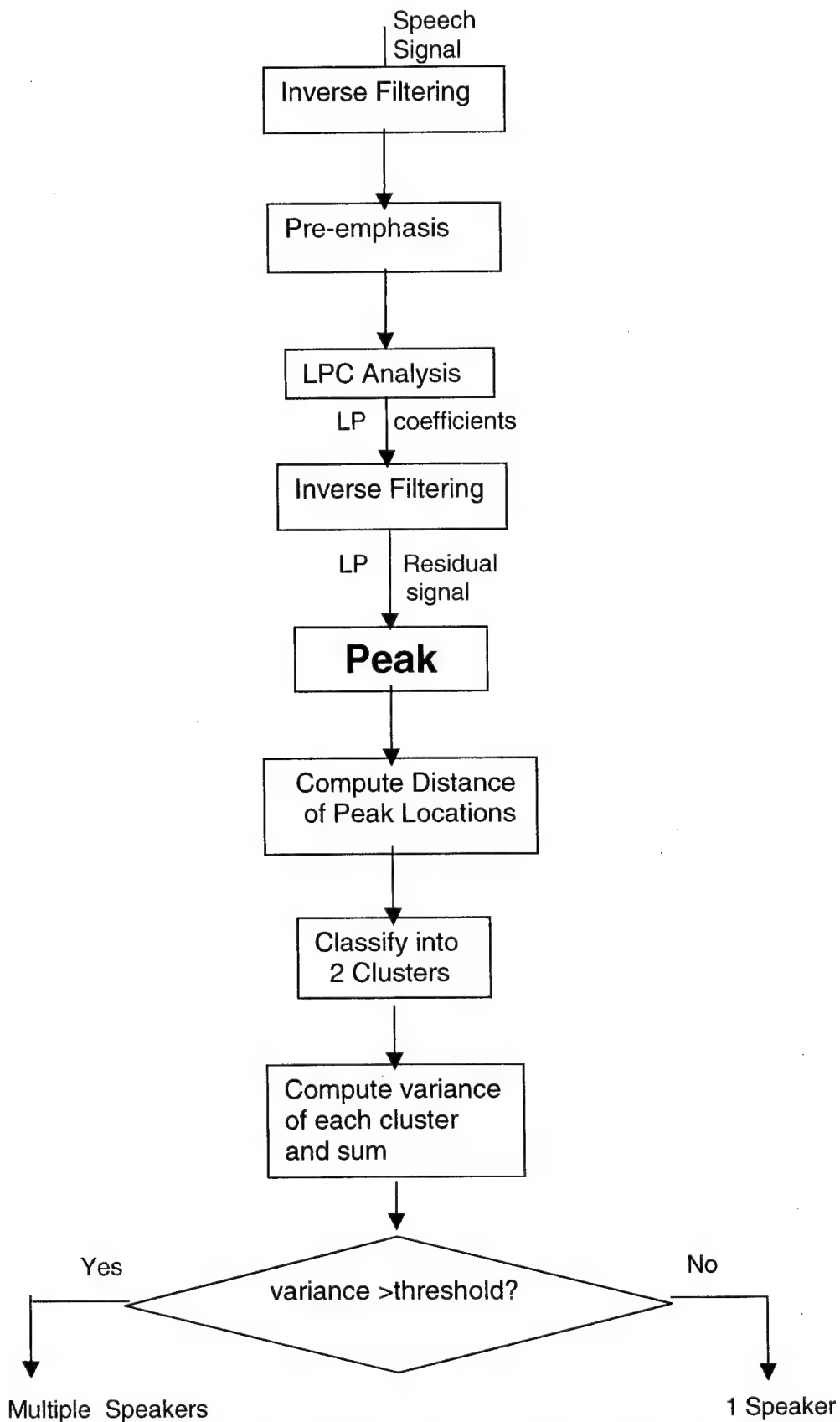


Figure 4.2. Speaker Count algorithm with LPC residual based pitch feature

4.2.1. Typical case

Female and male speaker signals are mixed together to generate a co-channel speech. In the residual of a single speaker, the variance of a distance between the residual peaks is small (as seen in bottom left pane showing pitch period histograms in Figure 4.2.1-a and Figure 4.2.1-b), while in the mixed speech residual, the same measure is large (Figure 4.2.1-c).

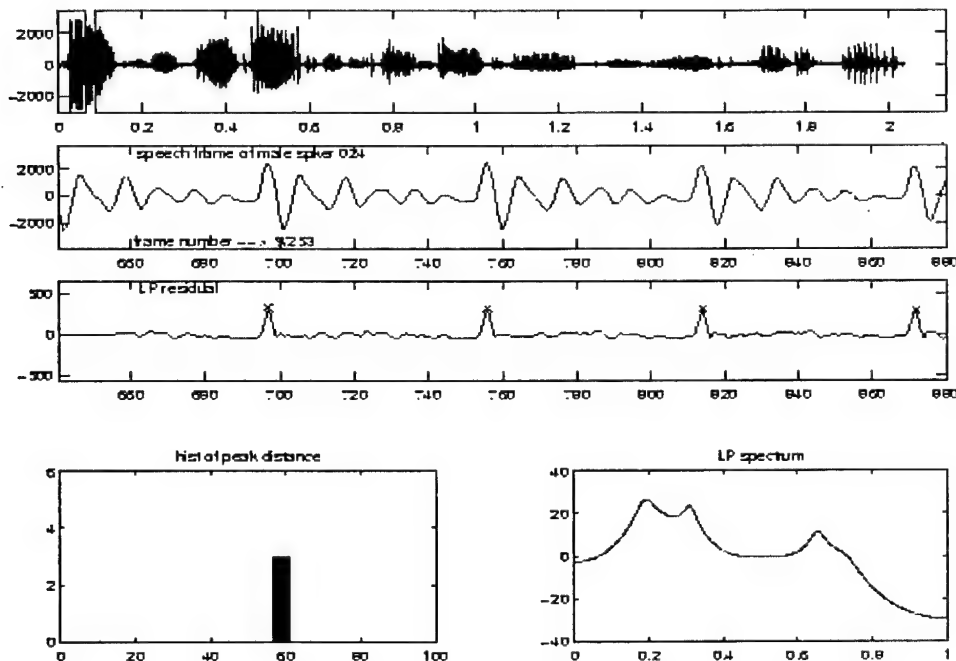


Figure 4.2.1-a. Male speech before mixing

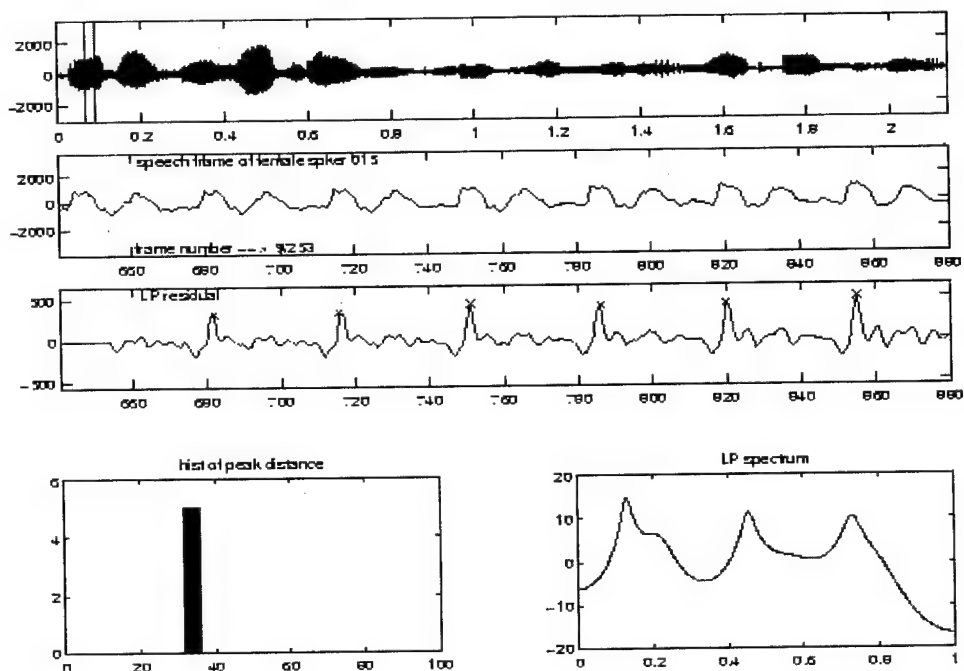


Figure 4.2.1-b. Female speech before mixing

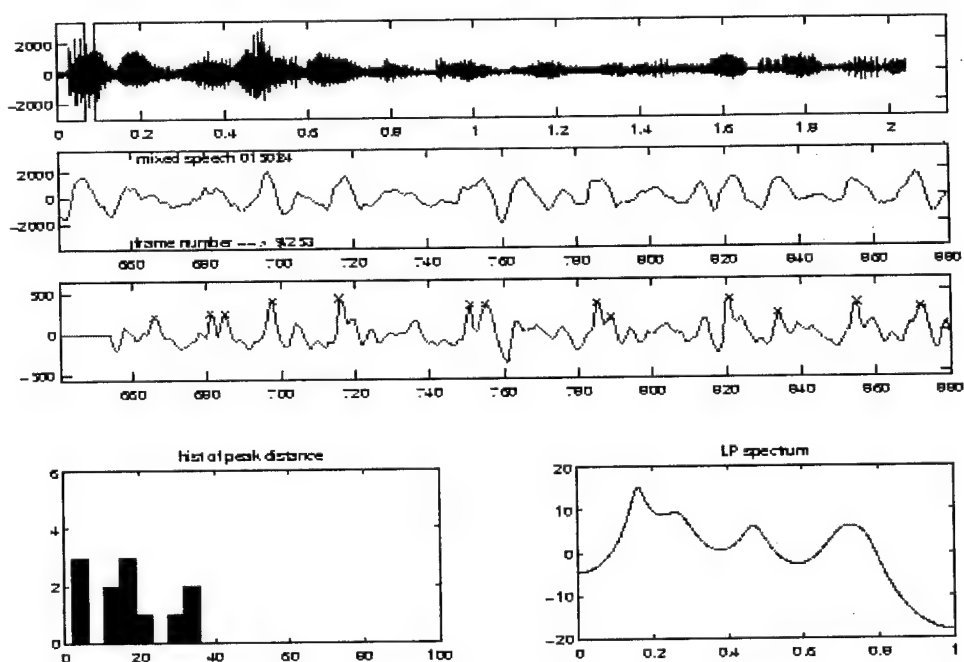


Figure 4.2.1-c. Mixed speech

4.2.2. Pitch Doubling

The residual of the male speaker in the following example indicates a “pitch doubling”. Pitch doubling occurs when an additional peak is detected almost in the middle of the typical pitch period. Since the estimated pitch period was grouped into two clusters, this phenomenon is accounted for, and, therefore, does not yield to an error (see bottom left pane showing pitch period histogram in Figure 4.2.2).

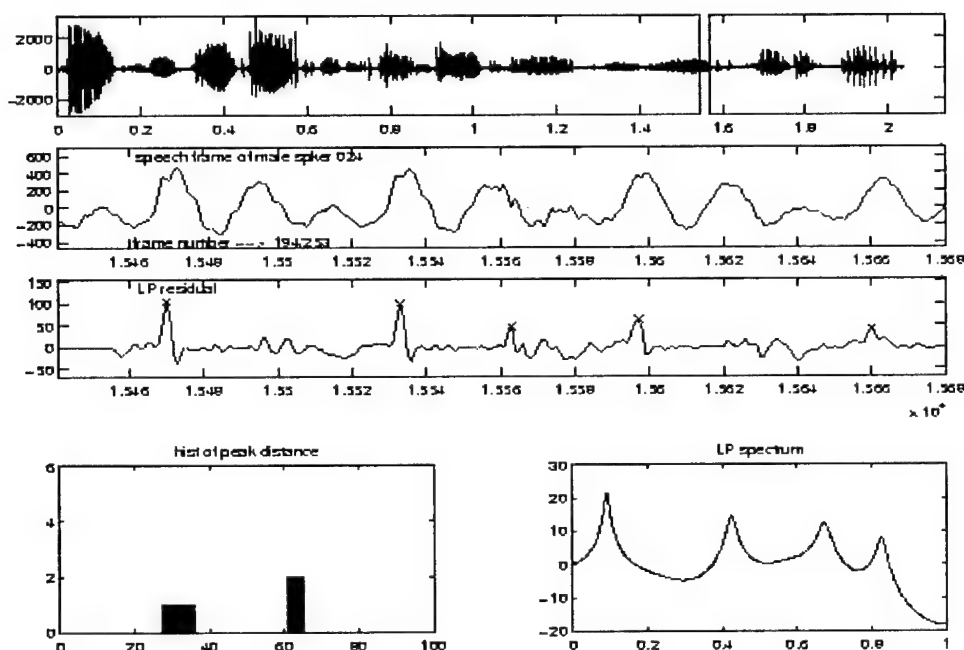


Figure 4.2.2. An example of pitch doubling

4.2.3. Pitch Halving

The residual of the female speaker in the following illustration indicates a “pitch halving”. Pitch halving occurs when a peak is missed in the LPC residual. The estimated pitch period was subdivided into two clusters, and, therefore, the pitch halving effect does not yield to an error (see bottom left pane showing pitch period histogram in Figure 4.2.3).

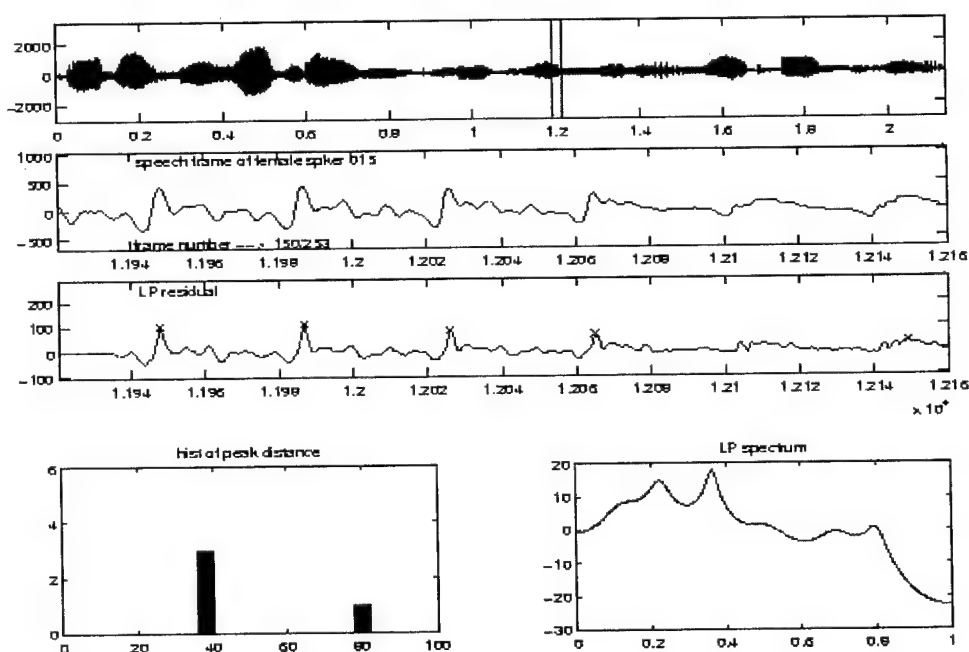


Figure 4.2.3. An example of pitch halving

4.2.4. Female speaker dominating

In cases where one speaker is dominating (higher signal energy compared to other speaker's signal), the peaks exhibit similar periodic structure as for single-speaker case and would typically be incorrectly classified by the algorithm. Figure 4.2.4-a shows the male speech with very low energy, which is dominated by the female speech in Figure 4.2.4-b. The resultant mixed speech in Figure 4.2.4-c has a very periodic pitch structure (as seen in the third pane in Figure 4.2.4-b).

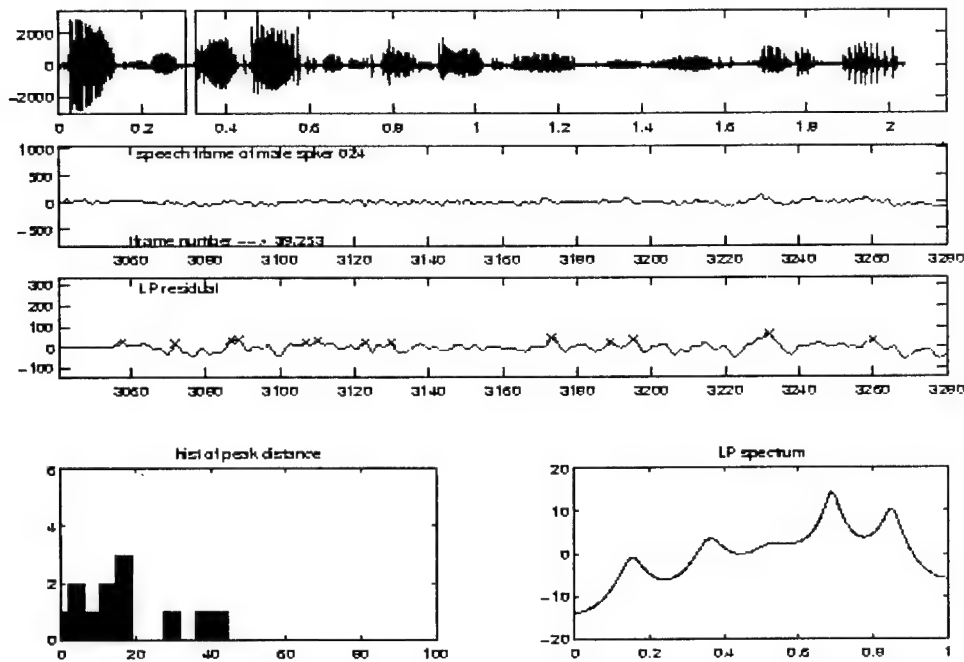


Figure 4.2.4-a. Male speech before mixing

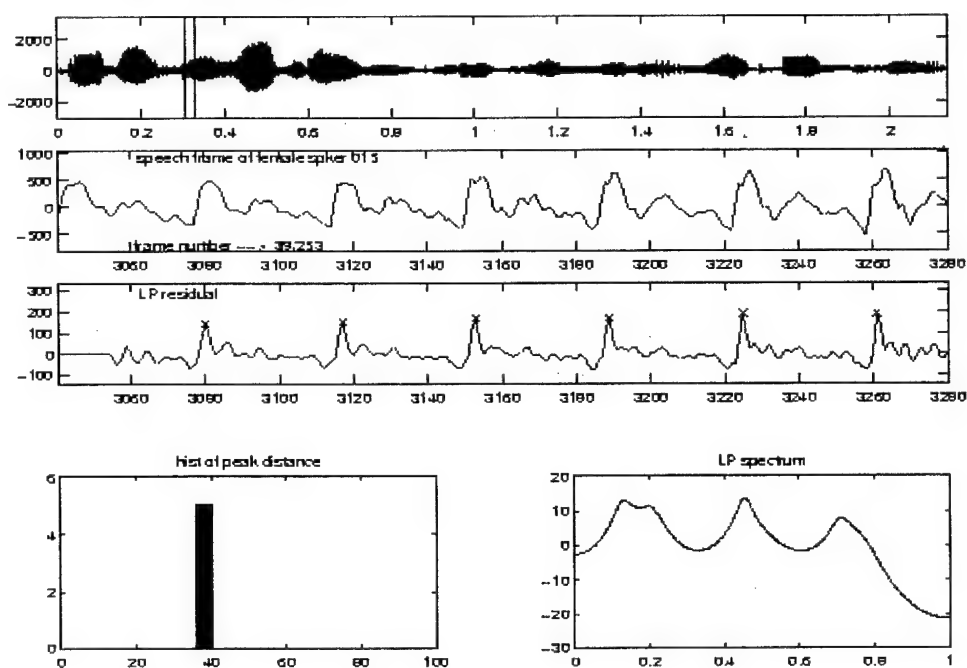


Figure 4.2.4-b. Female speech before mixing

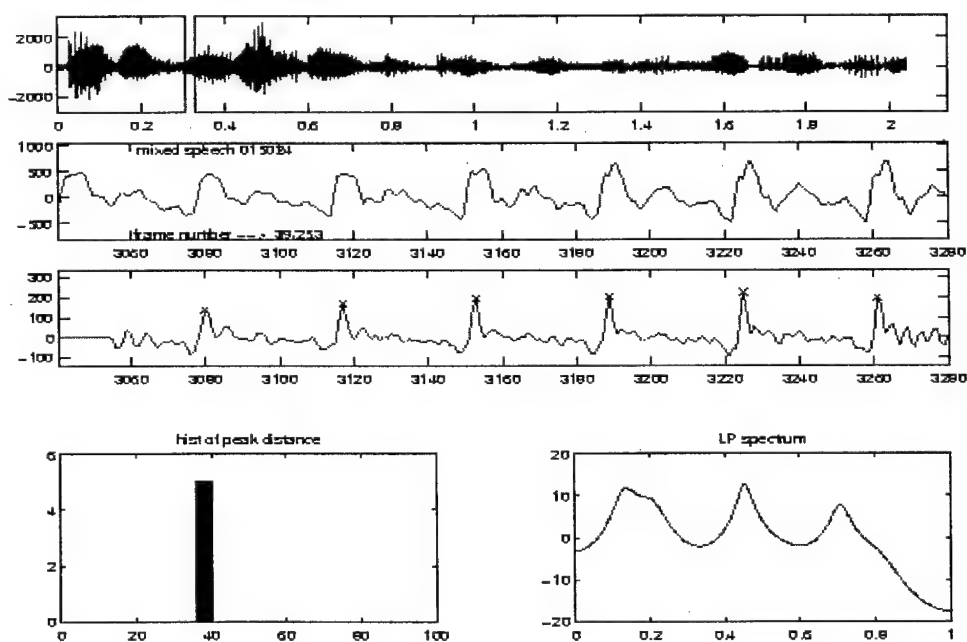


Figure 4.2.4-c. Mixed speech

4.2.5. Male speaker dominating

Similar to the earlier illustration, Figure 4.2.5-a shows a male speech, which dominates a female speech in Figure 4.2.5-b. A resultant mixed speech in Figure 4.2.5-c has a very periodic pitch structure (as seen in Figure 4.2.5-b) and may be mis-recognized by the algorithm.

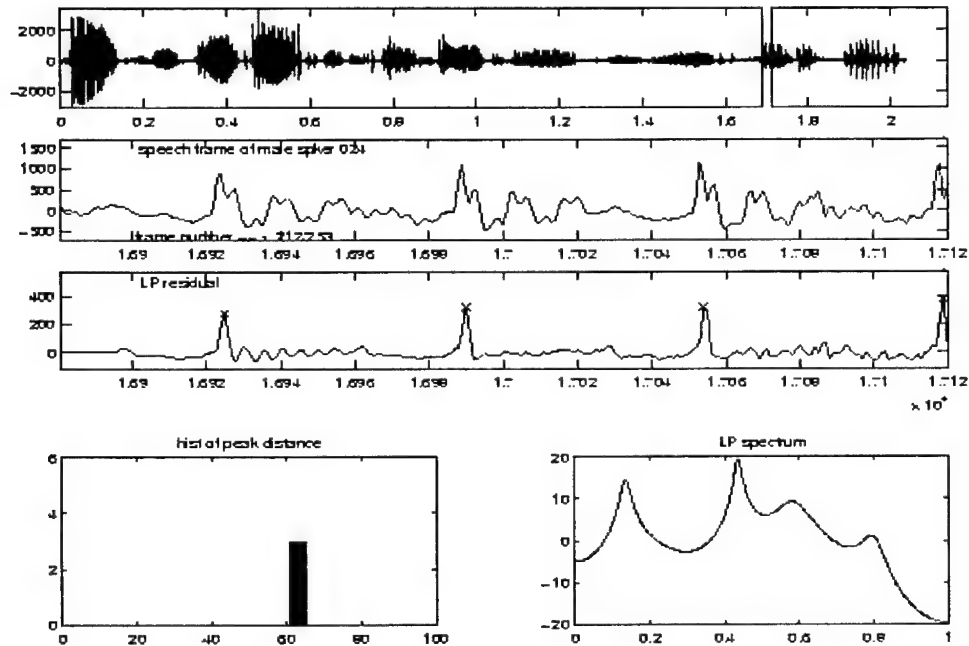


Figure 4.2.5-a. Male speech before mixing

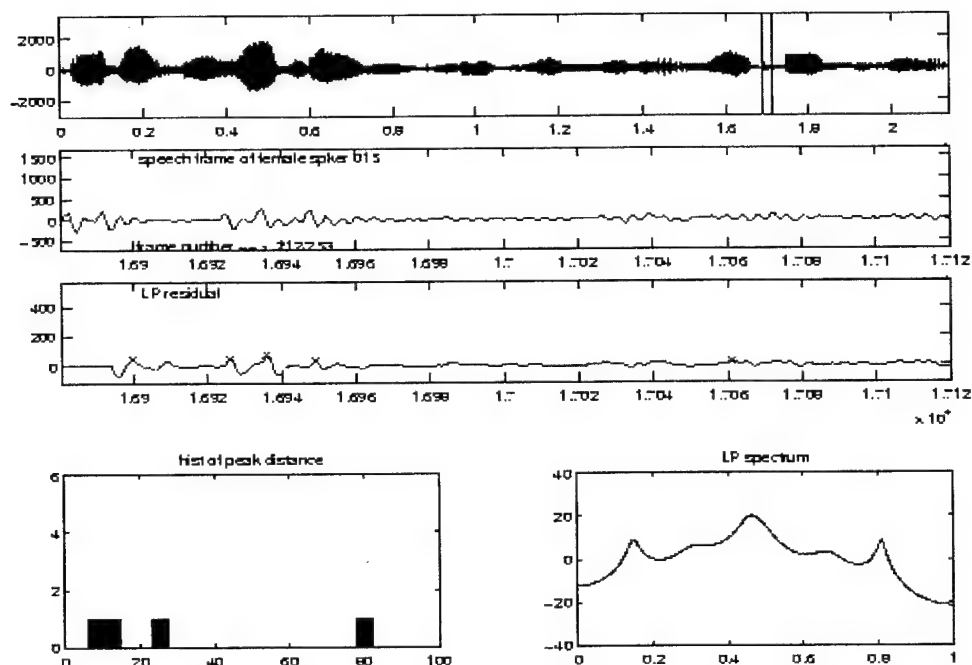


Figure 4.2.5-b. Female speech before mixing

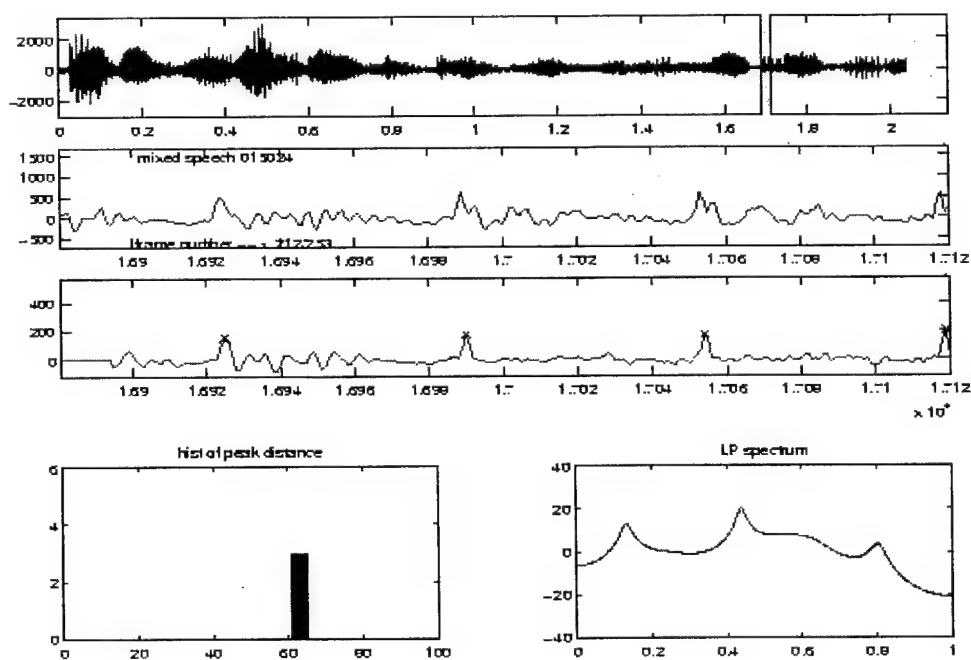


Figure 4.2.5-c. Mixed speech

he further investigation of residual based pitch variance algorithm aimed to analyze frame energy patterns.

4.2.6. Analysis of Energy and Energy Ratio

The analysis was done on the speech frame energy and speech frame energy ratio. It was found that for single speaker speech the algorithm is more likely to make an error when the frame energy is low, since it is difficult to pick up any periodic features from the LPC residual. For multiple speaker speech, on the other hand, the algorithm is more likely to classify the speech as a single-speaker speech when the energy ratio is low.

The energy ratio in each frame is defined as:

$$\text{energy_ratio} = \frac{\min(e_1, e_2)}{\max(e_1, e_2)},$$

where, e_1 and e_2 are frame energy of the two speakers mixed together, and is defined as:

$$e_1 = \frac{1}{N} \sum_j s_1^2(j) \quad \text{and} \quad e_2 = \frac{1}{N} \sum_j s_2^2(j),$$

where N is the number of samples in one frame, and s_i is the i -th speaker's speech signal.

A low energy ratio means that one speaker is dominating over the other. An energy ratio close to 1 means both speakers' signals have a similar energy.

Figure 4.2.6 shows three subplots. The top two subplots show the analysis of single-speaker speech files (female and male respectively). Each subplot gives:

- Pitch variance metric used to make speaker count decision,
- Final decision, i.e., the number of speakers present ("1" or "2" speakers), and
- Normalized frame energy.

As can be seen from the top two subplots, most of the errors (decision = 2, instead of 1) happen at those frames where the frame energy is really low.

The last subplot shows the speech file analysis for the mixed (co-channel) speech signal. The energy ratio plot is now added to the other three metrics being visualized. Most of the errors (decision = 1, instead of 2) occur at those frames where energy ratio is really low.

Table 4.2.6 below shows the system performance results – original ones and those when decision adjustments are made based on the frame energy values.

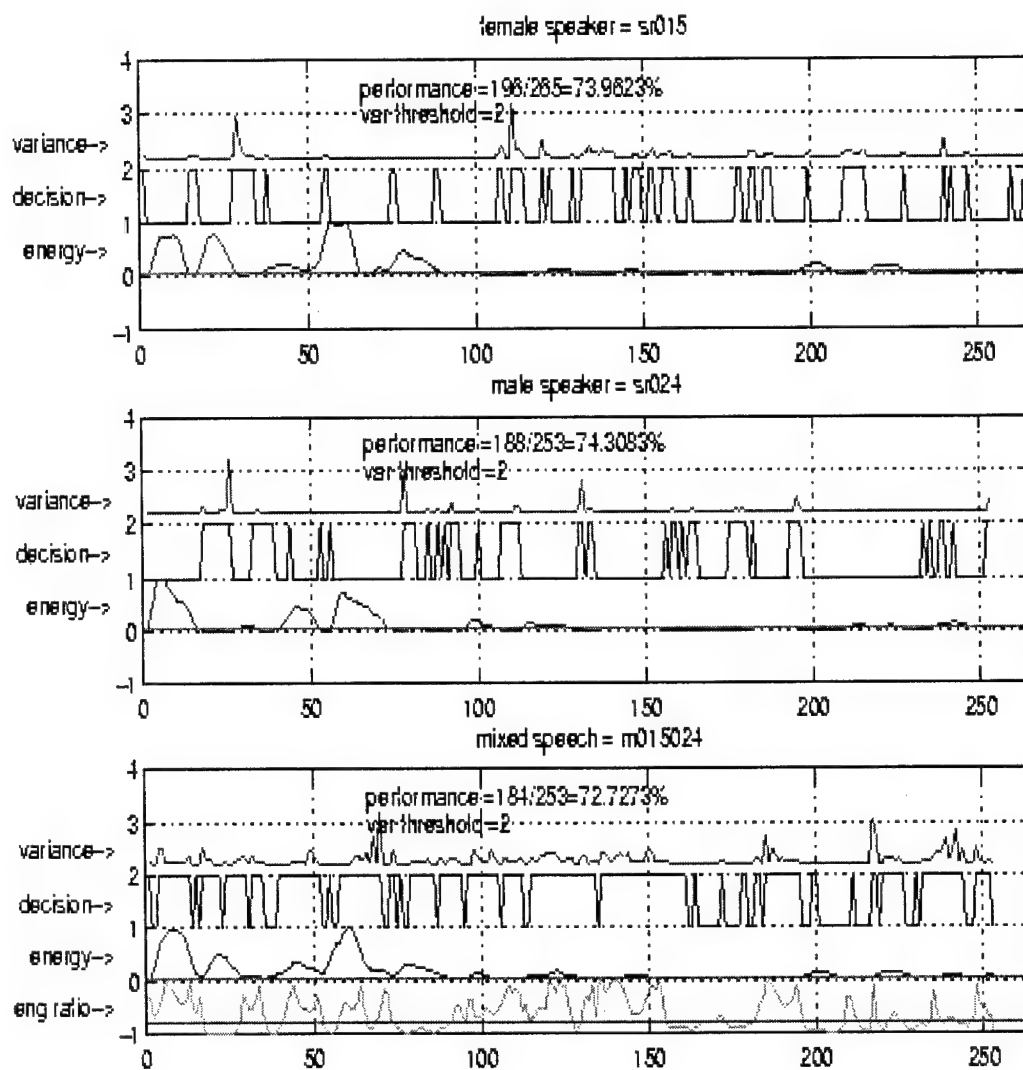


Figure 4.2.6. Performance, energy and energy ratio.

	Baseline Performance (all frames used) %	Performance When Frames with Energy < 0.01 thrown away %	Performance When decision=1 as a right decision when E ratio < 0.1 %
1 speaker	76.5%	82.1%	82.1%
2 speakers	63.4%	62.9%	83.6%
overall	65.4%	65.1%	83.2%

Table 4.2.6. LPC Residual Pitch based system performance where,

Second column – baseline result (variance threshold = 2),
 Third column – low energy frames (i.e., frame with energy < 0.01 maximum) are discarded. Affect 1-speaker result more than 2-speaker result.
 Fourth column – if reference decisions for all frames where one speaker is dominating (i.e., energy ratio < 0.1 (-10db)) are modified as single-speaker (“1”) label.

4.3. Experimental Setup

Experimental conditions and data were described in the section 3.2.2 “Baseline Experimental Setup”, Table 3.2.2. The same conditions were applied for pitch-based derived features, except for:

- Frame size parameter, which was set to 10 msec with no overlap for PPF feature,
- Voicing detection procedures were added to the training and testing phases for PPF feature. Only those speech frames, which were classified as “voiced” by the voicing detection module, were used during the training and testing phases,
- The variance threshold was set to 2.0 for LPC residual based pitch feature.

4.4. Experimental Results for PPF

Experiments with the pitch-based features were organized in the same manner as they were described in the previous chapter, where LP spectral-based features and different classifiers were tested.

4.4.1. System Performance with Vector Quantizer Classifier

The performance was evaluated as a function of framesize for 8 kHz sampling rate. The best results were with a framesize of 10 ms (see results in Table 4.4.1 below).

Codebook size	Pitch feature (PPF), %
1	57
2	43
4	50
8	49
16	52
32	53
64	53
128	52
256	54

Table 4.4.1. Performance for the PPF feature using a vector quantizer classifier

Observations

The codebook of size one gave the best result - of 57% success rate, which is close to the performance obtained with spectral features, while the other codebook sizes yield to those essentially.

4.4.2. System Performance with Neural Tree Network

Neural Tree Network classifier was tested with the PPF feature. The results are summarized in Table 4.4.2.

Number of levels	PPF, %
2	54
4	51
6	50
8	50
10	50

Table 4.4.2. Performance for the PPF feature using the neural tree network classifier.

Observations

The results obtained while utilizing NTN classifier yield to those obtained with VQ classifier.

4.4.3. System Performance on an Increased Number of Simultaneously Talking Speakers

Thirty-nine ("C39") instead of six speakers ("O6") were used for training and testing. The VQ codebook sizes 1 to 16 were tested. Speakers used for training and testing were the same, which defined the "closed-set" conditions. However, the utterances used for training and testing were different. Table 4.4.3 gives the results.

Codebook size	Original Test O6	39-speaker Test C39
1	57	60
2	43	60
4	50	56
8	49	47
16	52	54

Table 4.4.3. Performance for the PPF feature for original 6-speaker "open-set" tests (O6) versus current 39-speaker "closed-set" tests (C39).

Observations

The current results are comparable with obtained previously in the tests for 6 speakers.

The results of testing the PPF feature showed that the system performance was lower comparing to when LP spectral-based features were utilized, The further experiments, therefore, are to investigate how the system performs with LPC residual based pitch feature applied.

4.5. Experimental Results for LPC Residual Based Pitch

The following experiments were conducted to test the LPC Residual Based Pitch Algorithm:

1. System Performance with Baseline Parameters
2. Adding Voicing Detection to LPC Residual Based Pitch Algorithm
3. System Performance with Non-silence Speech
4. System Performance at Various Signal-to-Interference Ratios (SIR)
5. System Performance on an Increased Number of Simultaneously Talking Speakers
6. Study Correlation between System Decision and Frame Energy
7. Comparing Performance of the Speaker Count System on Different Speaker Gender Combination
8. New Approach for Speech-Silence Discrimination Error
9. New Silence Detection Algorithm
10. System Performance Under Different Pitch Variance Threshold
11. Classification of Low Energy Frames as Single Speaker Frames

4.5.1. System Performance with Baseline Parameters

Experimental Setup

In the experiments the baseline setup conditions were applied, as described in section 3.2.2 “Baseline Experimental Setup” to determine the LPC Residual Based Pitch algorithm performance, and compare it to previously obtained results.

Results

The results are provided in Table 4.5.1.

correct	Type 1 error	Type 2 error	Type 3 error
61.6%	12.5%	25.8%	0.1%

Table 4.5.1 System results for baseline testing.

Observations

The results of experiment show a couple percent of improvement compared to the result obtained from LPC based spectral features.

4.5.2. Adding Voicing Detection to LPC Residual Based Pitch Algorithm

This test was done to estimate the system performance when only voiced speech is considered. For calculating LP residual based pitch feature, eliminating the unvoiced frames will give the LP residual of all voiced speech frames.

Experimental Setup

The voicing detection was performed on mixed speech, and only the voiced speech frames were processed. The parameters were set up as described in section 3.2.2 "Baseline Experimental Setup".

Results

The results are shown in Table 4.5.2.

	correct	Type 1 error	Type 2 error	Type 3 error
Baseline	61.6%	12.5%	25.8%	0.1%
Voiced only	63.8%	8.4%	27.7%	0.03%

Table 4.5.2. Performance for voiced speech only

Observations

The system performance improves a couple percent when only the voiced frames were considered. The LP residual based pitch feature can be better detected.

4.5.3. System Performance with Non-silence Speech

To measure the performance of the system when silence are removed from single speech signals before mixing them to create multi-speaker speech file. Eliminating the silence frames prior to all other processing will give multispeaker speech without transitional single-to-multi-speaker and multi-to-single-speaker frames.

Experimental Setup

In this test all silence frames were eliminated from single speaker speech data prior to mixing them to create a multi-speaker speech file. Rest of the processing remains the same. The experimental setup parameters used as described in section 3.2.2 “Baseline Experimental Setup”.

Results

The results are shown in Table 4.5.3.

	correct	Type 1 error	Type 2 error	Type 3 error
Baseline	61.6%	12.5%	25.8%	0.1%
Non-silence only	52.3%	0%	47.7%	0.1%

Table 4.5.3. Performance for non-silence speech only

Observations

Removing the silence prior to mixing degraded the system performance by several percent.

4.5.4. System Performance at Various Signal-to-Interference Ratios (SIR)

Experimental Setup

For this experiment, the signals with consecutive steps of 6 dB were obtained by successive multiplication of one single-speaker speech signal by a weighting coefficient $\beta=0.5$ while mixing two single-speaker signals together. The weighting coefficient of the second single-speaker speech signal α remained the same and was equal to 0.5.

	0 dB,	6 dB,	-6dB,	12 dB,	18 dB,	24 dB,
α	0.5	0.5	0.25	0.5	0.5	0.5
β	0.5	0.25	0.5	0.125	0.0625	0.03125

Table 4.5.4. Ratios α and β at which single-speaker signals were combined

Results

The results were shown in Table 4.5.4.

0 dB, %	6dB, %	-6dB, %	12 dB, %	18 dB, %	24 dB, %
61.6%	59.3%	60.0%	56.0%	50.5%	46.0%

Table 4.5.4. Performance at different SIR ratios for pitch variance feature

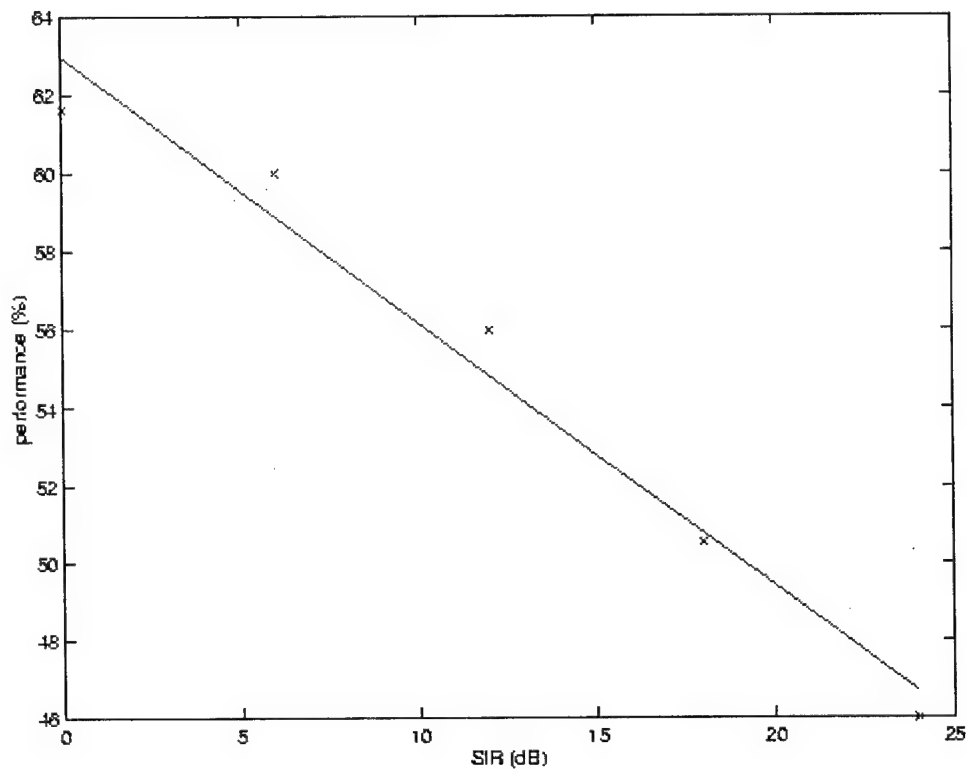


Figure 4.5.4. Performance at different SIR ratios

Observations

The performance of the system monotonically decreases as SIR ratio changes from 0 to 24dB.

4.5.5. System Performance on an Increased Number of Simultaneously Talking Speakers

To test the system performance on the speech where the number of speakers talking simultaneously is more than two.

Experiment Setup

Experiments were conducted using six speakers in an open-set environment (different speakers during training and testing). They were run with a co-channel speech obtained by mixing utterances from three speakers, four speakers and five speakers. For "3-speaker-test" five sentences of each speaker were taken to create multispeaker speech combinations. For "4-speaker-test" three different sentences of each speaker were taken to create multispeaker speech combinations. For "5-speaker-test" two sentences of each speaker were used. The rest of setup parameters was as described in section 3.2.2 "Baseline Experimental Setup".

Results

The results are shown in Table 4.5.5.

	correct	Type 1 error	Type 2 error	Type 3 error
2-speaker	61.6%	12.5%	25.8%	0.1%
3-speaker	66.0%	7.7%	26.2%	0.08%
4-speaker	69.7%	7.5%	22.8%	0.04%
5-speaker	82.0%	2.6%	15.5%	0%

Table 4.5.5. Results for different number of speakers present in multispeaker speech.

Observations

As the number of simultaneously talking speakers increased up to three, four and five (instead of two speakers used originally) the system's performance steadily improved.

4.5.6. Study Correlation between System Decision and Frame Energy

experiment was conducted in order to find the correlation between the frame labeling decision and the energy of that frame.

Experimental Setup

The experimental conditions in this experiment are the same as in baseline setting (see section 3.2.2 "Baseline Experimental Setup"). Each frame's energy was recorded (in dB). System decisions' distribution was analyzed as a function of the frame energy. The frame energy was normalized so that the maximum value is 1 (0 dB).

Results

Figure 4.5.6-1 to Figure 4.5.6-4 below show the distribution of correct decisions and type 1, 2, and 3 error as a function of frame energy.

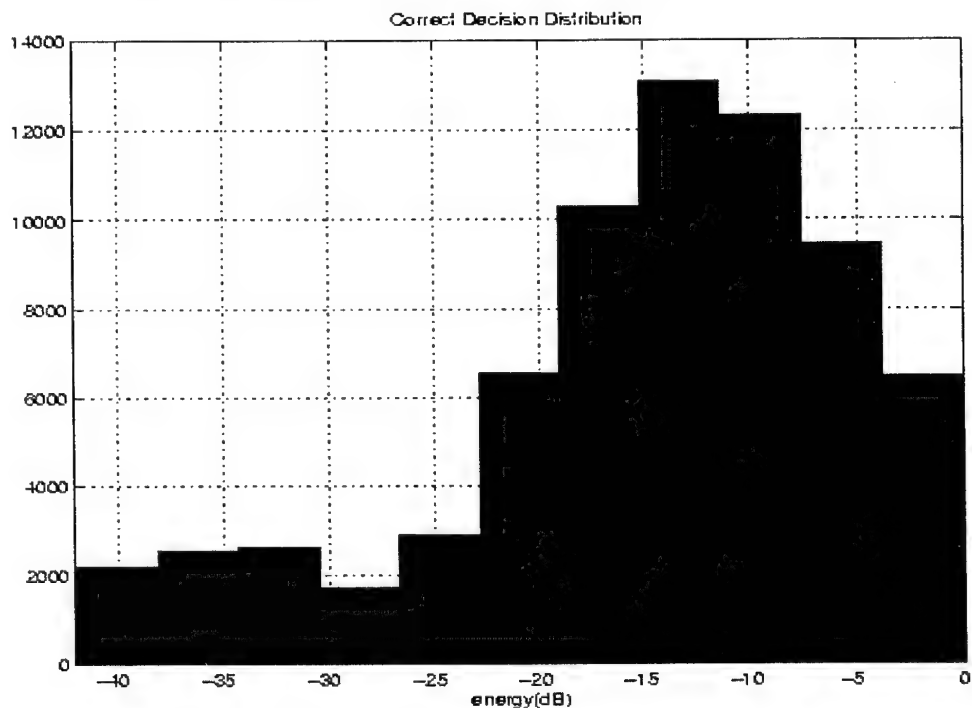


Figure 4.5.6-1 Distribution of correct system decision as a function of frame energy

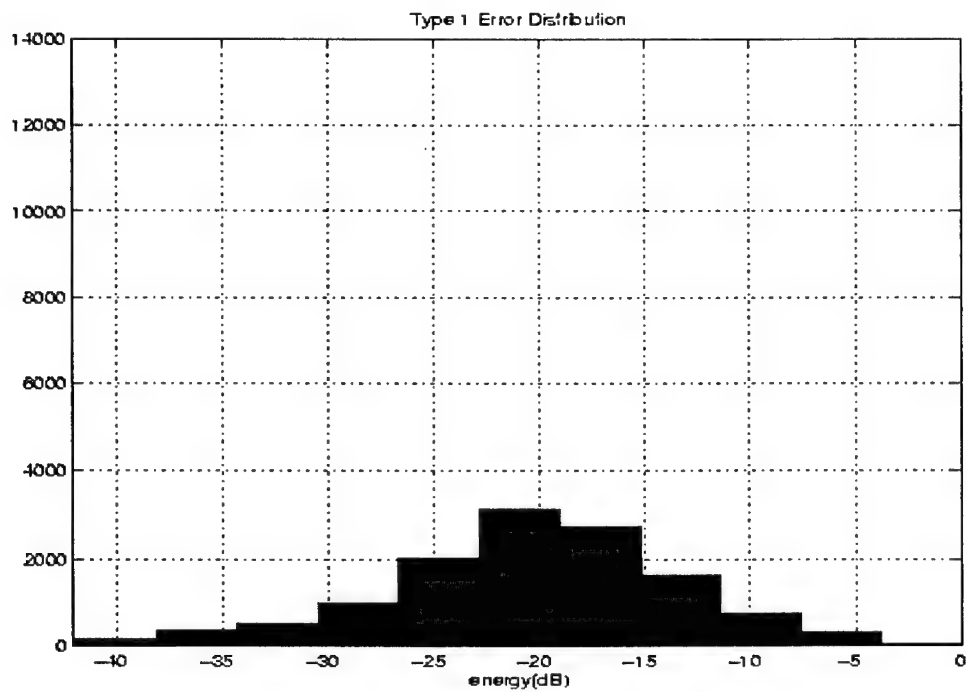


Figure 4.5.6-2 Distribution of type 1 error as a function of frame energy

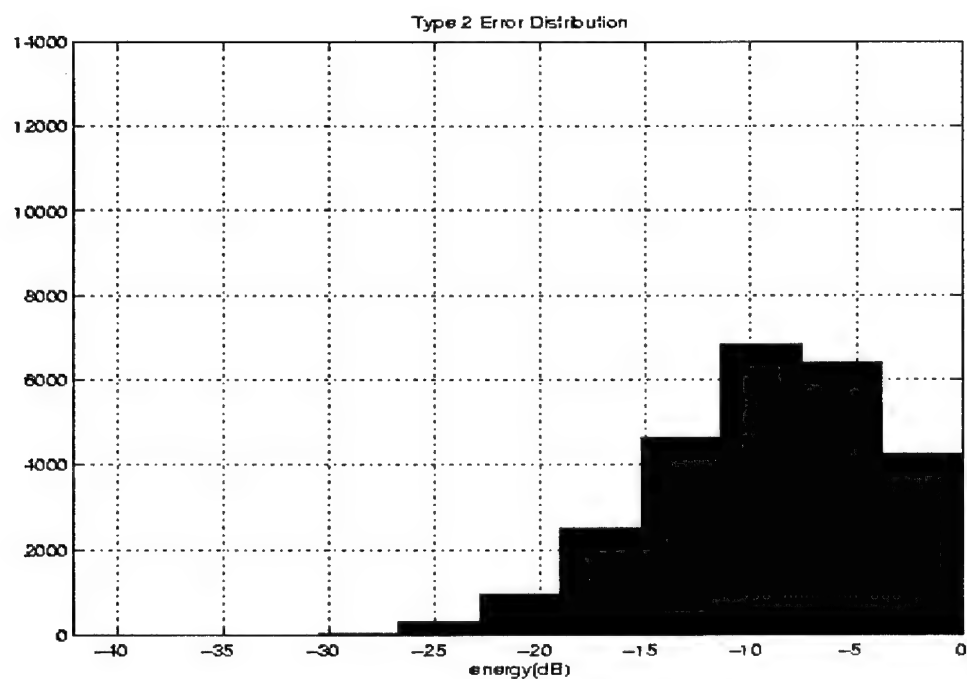


Figure 4.5.6-3 Distribution of type 2 error as a function of frame energy

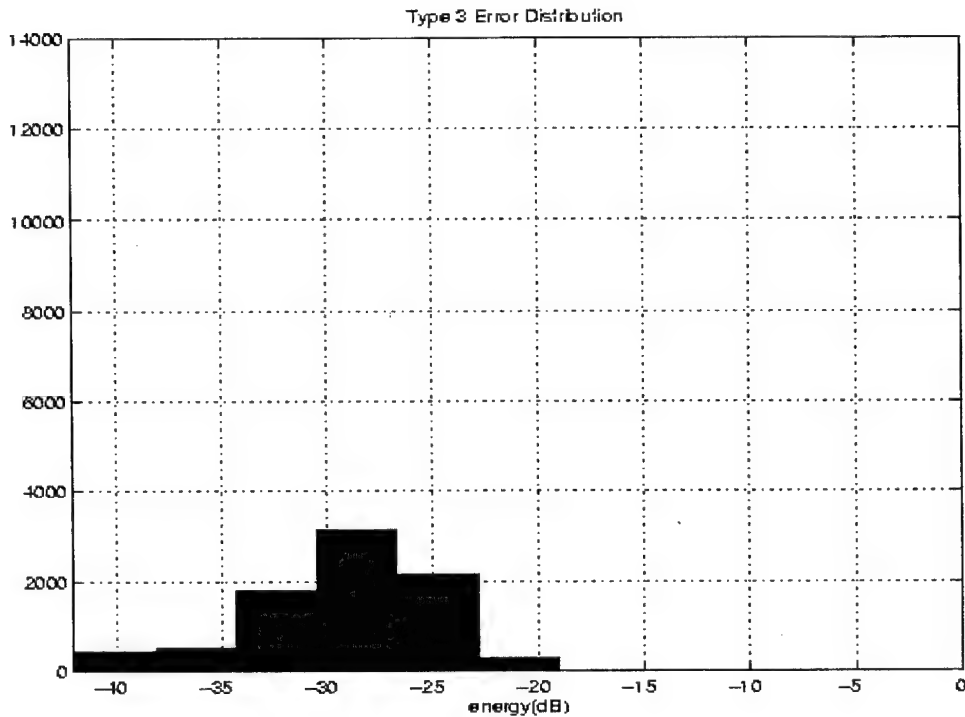


Figure 4.5.6-4. Distribution of type 3 error as a function of frame energy

Observations

As it is seen from the figures, type 1 errors center at about -20 dB, while type 2 errors center close to 0 dB (higher energy).

It could be concluded from the experiment, that by classifying the low energy frames as “1” speaker, type 1 error would decrease significantly, while type 2 error would increase only slightly, and the overall system performance should improve.

4.5.7. Comparing Performance of the Speaker Count System on Different Speaker Gender Combinations

To compare the system performance on different gender speaker speech versus both genders mixing speech, as it was done in prior experiments, this test was run.

Experimental Setup

In all the previous experiments an equal number of male (M) and female (F) speaker speech files were utilized for creating mixed, multispeaker speech files. Therefore, the resulting multispeaker files were of any possible combination: M + M, M + F, F + F. Here the experiments were separated into same gender (M+M and F+F) and different gender (M+F):

1. Same gender speech only – M + M and F+F;
2. Male speech + female speech combination – M + F;

In both experiments the baseline setup conditions were applied (see section 3.2.2 “Baseline Experimental Setup”).

Results

The results are provided in Table 4.5.7 for the above experiments.

	correct	Type 1 error	Type 2 error	Type 3 error
Same gender (M+M and F+F)	61.5%	12.3%	26.1%	0.1%
Different gender (M+F)	61.7%	12.5%	25.8%	0.1%

Table 4.5.7. System results for different speaker gender combinations.

Observations

The results of experiments show very similar performance, with slightly better result in the mix gender speech test.

4.5.8. New Approach for Speech-Silence Discrimination Error

The new approach for speech-silence discrimination error (type 3 error) calculations is described in subsection 3.3.14 of Chapter 3 for testing with LP cepstrum feature. In new type 3 error is redefined to occur when the reference is “*silence*” but estimated decision is “*speech (non-silence)*”, or when the reference is “*non-silence*” but estimated decision is “*silence*”.

Experimental Setup

The baseline setup conditions were applied (see section 3.2.2 “Baseline Experimental Setup”). The results provided in Table 4.5.8 compare the newly obtained performance with that obtained with the original Type 3 error calculation technique.

	Correct	Type 1 error	Type 2 error	Type 3 error
Old type 3 error	61.6%	12.5%	25.8%	0.1%
New type 3 error	64.9%	9.4%	19.4%	6.3%

Table 4.5.8. System results with a new approach for Type 3 error calculation

4.5.9. New Silence Detection Algorithm

New silence detection histogram-based algorithm is described in subsection 3.3.15 of Chapter 3, where it was tested with LP cepstrum feature.

Experimental Setup

The baseline setup conditions were applied: applied (see section 3.2.2 “Baseline Experimental Setup”). New Type 3 error calculation was performed.

Results

The results of the experiment are provided in Table 4.5.9 below.

	Correct %	Type 1 error %	Type 2 error %	Type 3 error %
Old Silence Detection Algorithm	64.9	9.4	19.4	6.3
New Silence Detection Algorithm	74.2	11.2	12.1	2.5

Table 4.5.9. System results with a histogram-based silence detection algorithm.

Observations

The results of experiments show about 10 percent improvement over previous best result. The speech-silence discrimination error (type 3) decreased, as expected from a more robust silence detection algorithm. Moreover, due to consistency in labeling speech-silence regions of speech during training and testing phases, the other error rates decreased as well.

4.5.10. System Performance Different Pitch Variance Threshold

To find the correlation between system performance and pitch variance threshold used to classify single and multiple speaker frames.

Experimental Setup

The experimental conditions in this experiment are the same as in baseline setting (see section 3.2.2 “Baseline Experimental Setup”).

Results

Different pitch variance classification thresholds were tested, and the correct decisions, Type 1, Type 2 and Type 3 errors were recorded. Type 3 error remained the same as it is not affected by the pitch variance threshold.

Table 4.5.10 shows the system performance, Type 1, Type 2 and Type 3 error with different variance threshold used. The percentage here was calculated using the new Type 3 error calculation (as described in section 4.5.8) approach.

	Correct %	Type 1 error %	Type 2 error %	Type 3 error %
Variance threshold=4.0	64.5	9.1	20.2	6.3
Variance threshold=3.5	64.6	9.2	20.0	6.3
Variance threshold=3.0	64.7	9.2	19.9	6.3
Variance threshold=2.5	64.8	9.2	19.7	6.3
Variance threshold=2.0 (baseline)	64.9	9.4	19.4	6.3
Variance threshold=1.5	65.0	10.0	18.7	6.3
Variance threshold=1.0	65.1	10.1	18.5	6.3
Variance threshold=0.5	65.3	10.5	18.0	6.3
Variance threshold=0	65.5	13.7	14.5	6.3

Table 4.5.10. Performance, Type 1, 2, 3 errors versus different variance threshold

Figure 4.5.10 below shows the plot of correct decisions and Type 1, 2, and 3 errors as a function of variance threshold.

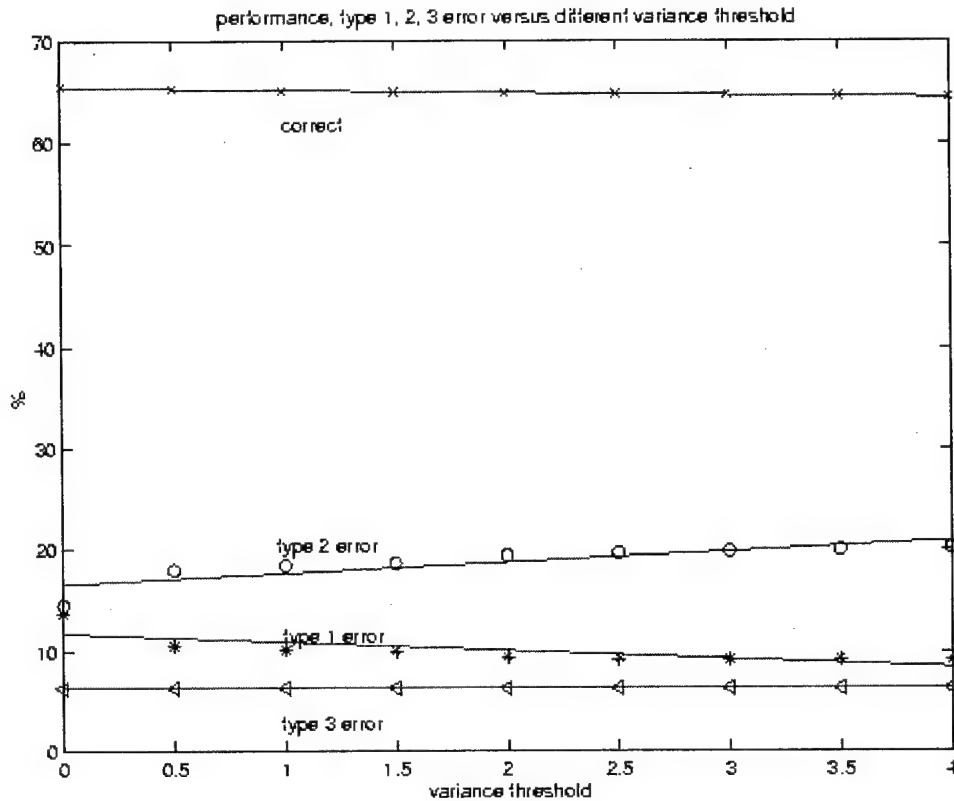


Figure 4.5.10. Performance, Type 1, 2, 3 errors versus different variance threshold

Observations

As the variance threshold increases, Type 1 error decreases, while Type 2 error increases at about the same rate. The overall system performance stays at about the same level.

4.5.11. Classification of Low Energy Frames as Single Speaker Frames

An analysis of the frame energy distribution (see subsection 4.5.6) showed that by classification of low energy frames as single speaker, it might be possible to essentially decrease type 1 error, while type 2 error would increase only slightly, leading to the overall improvement in system performance.

Experimental Setup

The experimental conditions in this experiment were the same as in the baseline setting (see section 3.2.2 "Baseline Experimental Setup"). Two experiments were run: with energy thresholds equal to -20 dB and -15 dB.

Frames with an energy less than the energy threshold were classified as single speaker frame.

Results

The results are provided in Table 4.5.11. The new speech-silence discrimination error (type 3) calculation (as described in section 4.5.8) was used.

	Correct %	Type 1 error %	Type 2 error %	Type 3 error %
Baseline (with new type 3 error)	64.9	9.4	19.4	6.3
New silence detection	74.3	11.2	12.1	2.5
New silence detection and Low energy as 1 (<-20dB)	78.2	6.6	12.8	2.5
New silence detection and Low energy as 1 (<-15dB)	79.2	2.8	15.6	2.5

Table 4.5.11. System results with classification of low energy frames as "1" speaker.

Observations

As expected, the results of experiments show several percent of improvement over the previous best results.

5. Experiments on Green Flag Database

The "Green Flag" speech database was made available as the test data set for the contract. Green Flag contains actual real life, ground-to-air communication speech. The speech files made available were primarily single speaker, and were stored as 16-bit linear PCM, 8 kHz sampled. Multiple talker speech data was simulated by additive mixing of given speech files. The database contained sixteen (16) speakers with twenty (20) utterances from each speaker. Each speaker utterances were mixed with the next speaker's utterances, giving in total 15 co-channel speech combinations x 20 utterances = 300 utterances for training and testing.

5.1. System Performance Benchmarks

In order to benchmark the system performance on the Green Flag test database, all parameters were used with the same settings as were set up for baseline conditions in earlier experiments with TIMIT database (Table 3.2.2 of section "Baseline Experimental Setup"). The following variations over the baseline settings were found to be of interest from earlier experiments, and therefore used in the benchmark process.

- LP cepstrum feature / VQ classifier / histogram-based silence detection (similar to system configuration in section 3.3.15)
- Residual-based pitch feature / histogram-based silence detection (similar to system configuration in section 4.5.9)
- LP cepstrum feature / VQ classifier / rule-based silence detection (similar to system configuration in section 3.3.1.1)
- Residual-based pitch feature / rule-based silence detection (similar to system configuration in section 4.5.4)

The same experiments have been conducted with the TIMIT database speech files, so the results obtained for both databases are compared directly in Table 5.1 below.

Experiment	TIMIT database		GREEN FLAG database	
	Correct	Er1/Er2/Er3	Correct	Er1/Er2/Er3
	%		%	
1. LP cepstrum /VQ/ histogram-based silence detection	68	15/15/2	60	16/9/15

2. Residual-based pitch/ histogram-based silence detection	74 14/10/2	48 37/0.5/15
3. LP cepstrum/VQ/rule- based silence detection	51 8/18/11	53 23/17/6
4. Residual-based pitch/ rule-based silence detection	47 23/19/11	45 48/1/6

Table 5.1 System performance on TIMIT and Green Flag databases speech files.

Observations

Pitch based algorithm degrades very quickly in noisy environments, and LP cepstrum feature-based algorithm outperforms pitch-based for Green Flag database.

5.2. System Performance Using Different Pitch Variance Threshold

To find the correlation between system performance and the pitch variance classification threshold used.

Experimental Setup

The experimental conditions in this experiment are the same as in baseline setting (see section 3.2.2 “Baseline Experimental Setup”). Different pitch variance classification thresholds were tested, and the correct decisions, Type 1, Type 2 and Type 3 errors were recorded. The pitch variance obtained from a frame of speech is compared against a pre-determined variance threshold for the classification decision of single or multiple speakers.

Results

Table 5.2 shows the system performance, Type 1, Type 2 and Type 3 error with different variance threshold used. The percentage here was calculated using the new Type 3 error calculation approach.

	Correct %	Type 1 error %	Type 2 error %	Type 3 error %
Variance threshold=4.0	48.0	36.5	0.5	15.1
Variance threshold=3.5	47.9	36.5	0.5	15.1
Variance threshold=3.0	47.9	36.5	0.5	15.1
Variance threshold=2.5	47.9	36.6	0.4	15.1
Variance threshold=2.0 (baseline)	47.8	36.7	0.4	15.1
Variance threshold=1.5	47.6	36.9	0.4	15.1
Variance threshold=1.0	47.7	36.8	0.4	15.1
Variance threshold=0.5	47.5	37.0	0.4	15.1
Variance threshold=0	47.2	37.4	0.4	15.1

Table 5.2. Performance, Type 1, 2, 3 errors versus different variance threshold

Observations

The system performance and operating point (type-1 and type-2 error distribution) does not change while varying the variance threshold values. The Green Flag database is quite noisy, resulting in a noisy LPC residual signal and spurious "pitch" peaks. The pitch variance in the Green Flag database is very high (even for single speaker speech), and the tested range of variance threshold has no affect in changing the operating point of the system.

6. Summary and Conclusion

6.1. Key Observations

The following key observations can be logged as lessons learned during the project:

- Removal or non-removal of silence in single speaker speech recordings before simulating a co-channel speech mixing has no significant effect on system accuracy;
- System performance is quite similar across different speaker gender mixes (same or cross gender mixing) in co-channel speech;
- System accuracy degrades several percent (~4-6%) for each 6 dB decrease in SIR ratio; Co-channel speech detection accuracy is same for +x dB or -x dB SIR;
- Co-channel speech detection accuracy improves with increasing number of simultaneous talkers;
- Transitional speech frames (single-to-multi speaker or vice-versa, speech-to-silence or vice-versa) are more prone to error, but still are not pre-dominant contributors to overall system error;
- No significant dependence of analysis frame size (80 to 480 milliseconds) on accuracy was detected;
- The overall system accuracy of the proposed algorithms can be summarized as follows:

“clean” speech database:	spectral features: ~ 69 %
	pitch based features: ~ 79 %
“noisy” speech database:	spectral features: ~ 60 %
	pitch based features: ~ 48 %
- Pitch based algorithm out-performed spectral based algorithm in “clean” speech environment;
- Pitch based algorithm degrades very quickly in noisy environments.

6.2. Future Research

- Perform robustness study on pitch-based algorithms;
- Compile co-channel test database with actual “ground truth”;
- Study actual benefit of speaker count determination as a pre-processor to speech and speaker recognition systems.

References

- [Assaleh94] K. T. Assaleh and R. J. Mammone, "New LP-derived features for speaker identification," *IEEE Trans. on Speech and Audio Proc.*, vol. 2, pp. 630-638, Oct. 1994.
- [Atal74] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *J. Acoust. Soc. Am.*, vol. 55, pp. 1304--1312, June 1974.
- [Davis80] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoustics, Speech Signal Proc.*, vol. 28(4), pp. 357-366, August 1980.
- [Deller93] J. R. Deller, J. G. Proakis and J. L. Hansen, *Discrete Time Processing of Speech Signals*, MacMillan, New York, 1993.
- [Duda73] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, Wiley, 1973.
- [Farrell94] K. R. Farrell, R. J. Mammone and K. T. Assaleh, "Speaker recognition using neural tree networks and conventional classifiers," *IEEE Trans. on Speech and Audio Proc.*, vol. 2, pp. 194-205, January 1994.
- [Frasier76] R.H. Frasier, S. Samsam, L.D.Briada, and A.V. Oppenheim, "Enhancement of speech by adaptive filtering," *IEEE Int. Conf. Record on Acoust., Speech and Sig. Proc.*, pp. 251-253, April 1976.
- [Hanson84] B.A. Hanson and D.Y. Wong "The harmonic magnitude suppression technique for intelligibility enhancement in the presence of interfering speech," *IEEE Int. Conf. Record on Acoust., Speech and Sig. Proc.*, pp. 18A.5.1-18.A.5.4, March 1984.
- [Juang87] B. H. Juang, L. R. Rabiner and J. G. Wilpon, "On the use of bandpass liftering in speech recognition," *IEEE Trans. on Acoust., Speech and Sig. Proc.*, vol. ASSP-35, pp. 947-954, July 1987.
- [Linde80] Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. on Comm.*, vol. COM-28, pp. 84-95, Jan. 1980.
- [Min88] K. Min, D. Chien, S. Li, and C. Jones, "Automated two speaker separation system," *IEEE Int. Conf. Record on Acoust., Speech and Sig. Proc.*, pp. 537-540, April 1988.

[Markel76] J. D. Markel and A. H. Gray, *Linear Prediction of Speech*, Springer-Verlag, New York, 1976.

[Oppenheimer89] A. V. Oppenheim, R. W. Schafer, *Discrete-Time Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[Paliwal82] K. K. Paliwal, "On the performance of the quefrency-weighted cepstral coefficients in vowel recognition," *Speech Communication*, vol. 1, pp. 151--154, May 1982.

[Parson76] T.W. Parsons, "Separation of speech from interfering speech by means of harmonic selection," *J. Acoust. Soc. Am.*, vol. 60(4), pp.911-918, October 1976

[Rabiner78] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, 1978.

[Rabiner93] L. R. Rabiner and B. Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, 1993.

[Ramachandran89] R. P. Ramachandran and P. Kabal, "Pitch prediction filters in speech coding", *IEEE Trans. on Acoust., Speech and Signal Proc.*, vol. ASSP-37, pp. 467-478, April 1989.

[Ramamoorthy88] V. Ramamoorthy, N. S. Jayant, R. V. Cox and M. M. Sondhi, "Enhancement of ADPCM speech coding with backward adaptive algorithms for postfiltering and noise feedback," *IEEE Jour. on Select. Areas in Commun.*, vol. 6, pp. 364-382, Feb. 1988.

[Rosenberg87] A. E. Rosenberg and F. K. Soong, "Evaluation of a vector quanization talker recognition system in text independent and text dependent modes," *Comp. Speech and Lang.*, vol. 22, pp. 143-157, 1987.

[Sankar93] A. Sankar and R. J. Mammone, "Growing and pruning neural tree networks," *IEEE Trans. on Computers*, vol. C-42, pp. 221--229, March 1993.

[Stevens57] S. S. Stevens, "Critical bandwidth in loudness summation," *J. Acoust. Soc. Am.*, 29:548-557, 1957.

[Sugamura81] M. Sugamura and F. Itakura, "Speech data compression by LSP analysis-synthesis technique", *Trans. of the Institute of Electronics, Information, and Computer Engineers*, vol. J64-A, pp. 599-606, 1981.

[Tohkura87] Y. Tohkura, "A weighted cepstral distance measure for speech recognition," *IEEE Trans. on Acoust., Speech and Sig. Proc.*, vol. ASSP-35, pp. 1414--1422, Oct. 1987.

[Wang92] Shiua Wang, A. Sekey, A. Gersho, "An objective measure for predicting subjective quality of speech coders," *IEEE Journal on selected areas in communication*, Vol. 10, No.5, June 1992.

[Zilovic] M. S. Zilovic, R. P. Ramachandran and R. J. Mammone, "Speaker identification based on the use of robust cepstral features obtained from pole-zero transfer functions," submitted to *IEEE Trans. on Speech and Audio Proc.*

[Zilovic97] M. S. Zilovic, R. P. Ramachandran and R. J. Mammone, "A fast algorithm for finding the adaptive component weighted cepstrum for speaker recognition," *IEEE Trans. on Speech and Audio Proc.*, vol. 5, pp. 84--86, Jan. 1997.

[Zissman90] M. A. Zissman and C. J. Weinstein, "Automatic Talker Activity Labeling For Co-Channel Talker Inteference Suppression," *CASSP '90 Proceedings*, vol.2, pp. 813-816. 1990.

[Zissman91] M. A. Zissman, "Cochannel talker interference suppression," Technical Rep.895, Lexington, Mass.: MIT Lincoln Laboratory, July 1991.

[Zissman92] M. A. Zissman and D.C. Seward IV, "Two-talker pitch tracking for co-channel talker interference suppression," Technical Rep.951, Lexington, Mass.: MIT Lincoln Laboratory, April 1992.

***MISSION
OF
AFRL/INFORMATION DIRECTORATE (IF)***

The advancement and application of information systems science and technology for aerospace command and control and its transition to air, space, and ground systems to meet customer needs in the areas of Global Awareness, Dynamic Planning and Execution, and Global Information Exchange is the focus of this AFRL organization. The directorate's areas of investigation include a broad spectrum of information and fusion, communication, collaborative environment and modeling and simulation, defensive information warfare, and intelligent information systems technologies.